

Universidade Federal de Santa Catarina

Programa de Pós-Graduação em Engenharia de Produção

**Monitoramento de fontes de informação na Internet: modelo  
multiagentes para suporte  
ao processo de Inteligência Competitiva.**

**Alexandre Faria Lemos**

Florianópolis  
2005

**Alexandre Faria Lemos**

**Monitoramento de fontes de informação na Internet: modelo  
multiagentes para suporte  
ao processo de Inteligência Competitiva.**

Dissertação apresentada ao Programa de  
Pós-Graduação em Engenharia de Produção  
da Universidade Federal de Santa Catarina,  
para a obtenção do grau de mestre em  
Engenharia de Produção.

Orientador: Pedro Felipe de Abreu

Florianópolis  
2005

ALEXANDRE FARIA LEMOS

**Monitoramento de fontes de informação na Internet: modelo multiagentes para suporte ao processo de Inteligência Competitiva.**

Esta dissertação foi julgada e aprovada para a obtenção do título de **Mestre em Engenharia de Produção** no Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de Santa Catarina.

Florianópolis, 21 de março de 2005.

---

Prof. Edson Pacheco Paladini, Ph.D.  
Coordenador do Curso

**Banca Examinadora:**

---

Prof. Pedro Felipe de Abreu, Ph.D.  
Orientador

---

Prof. Marcelo Thiry Comicholi da Costa, Dr.

---

Profa. Helena Pereira da Silva, Dr.

**Dedico este trabalho aos meus pais,  
Roberto Lemos e Maria Izolda de Faria  
Lemos, que com doses extremas de  
dedicação altruísta, paciência, sabedoria  
e amor, ao longo de uma vida inteira,  
transformaram uma criança em um  
homem adulto.  
Vocês são um modelo para toda a  
humanidade.**

**Agradeço infinitamente com muito amor.**

## **Agradecimentos**

Aos meus pais, Roberto Lemos e Maria Izolda de Faria Lemos, por proporcionarem condições, não somente materiais, mas especialmente intelectuais e emocionais, para o meu desenvolvimento e a realização deste trabalho. Jamais poderei agradecer o suficiente.

À minha irmã, Roberta Faria Lemos, por dar apoio nas horas que precisei, por dar estímulo nas horas que fraquejei e pelas broncas nas horas que mereci, sempre de forma altruísta e amorosa.

A Alessandra de Souza Borges, grande amor em minha vida, pelo apoio, energia e dedicação e por acreditar em mim quando eu mais precisava. Seu amor foi meu combustível, sua energia minha luz em momentos sombrios. Obrigado por tudo.

Ao meu orientador, Pedro Felipe de Abreu, por acreditar em minha capacidade, por ajudar quando precisei e por proporcionar a base sobre a qual me apoiei. Por ser uma pessoa de capacidade intelectual e retidão moral que me inspiram, e por me presentear com sua amizade.

A Aline França de Abreu, pelas conversas que ajudaram a solucionar dificuldades e pela extraordinária capacidade de fazer conceitos complexos parecerem claros. Sua amizade, e de sua família, me é muito cara.

A Dorzeli Salete Trzeciak, pela amizade e apoio, pela consideração e por compartilhar seus conhecimentos e sua sabedoria.

A todos os membros do IGTI, que fazem do Núcleo de Estudos em Inovação, Gestão e Tecnologia de Informação um ambiente de trabalho no qual a cooperação e a amizade são a regra e não a exceção.

A todas as pessoas que desenvolveram o OpenOffice.org, por proporcionar, gratuitamente, o software no qual este trabalho foi escrito.

## **Resumo**

Este trabalho consiste da proposta de um modelo para uma arquitetura de agentes inteligentes para uso como ferramenta no processo de inteligência competitiva. O trabalho inicialmente explora os conceitos de competitividade e inteligência competitiva, chegando na necessidade de uma ferramenta de apoio ao processo na fase de coleta de informações, especialmente na Web. A base para o desenvolvimento deste trabalho é a proposta apresentada em tese de doutorado por Silva (2000). É realizado então uma revisão do tema de agentes inteligentes e técnicas de análise de textos. Após é desenvolvida a proposta do modelo, uma proposta teórica que explora principalmente as relações entre as necessidades de inteligência competitiva e as características que deve apresentar a arquitetura de agentes inteligentes.

## **Abstract**

This work consists in a proposal of an intelligent agent architecture to use as a tool in the process of competitive intelligence. Initially, the work explores the concepts of competitiveness and competitive intelligence, arriving in the need to a support tool in the process, in the information collection phase, specially from the Web. The base for the development of this work is the proposal presented in a doctoral thesis by Silva (2000). A revision of the intelligent agent and text mining techniques and theories is then made. Then the proposal for the model is made, a theoretical proposal that explores mainly the relationships between competitive intelligence needs and the architecture's characteristics.

# Sumário

1	Introdução.....	11
1.1	Objetivos Geral e Específicos.....	12
1.2	Justificativa e Motivação.....	13
1.3	Delimitação.....	14
1.4	Natureza do trabalho e metodologia.....	15
1.5	Estrutura da dissertação.....	16
2	Competitividade e Inteligência Competitiva.....	17
2.1	Competitividade.....	17
2.2	Definição de Inteligência Competitiva.....	20
2.3	Importância da Inteligência Competitiva.....	21
2.4	Processo de inteligência competitiva.....	23
2.5	Inteligência competitiva e a internet.....	27
2.6	Inteligência competitiva, agentes e Internet.....	30
3	Agentes inteligentes e mineração de textos.....	33
3.1	Definição.....	33
3.2	Propriedades.....	35
3.2.1	Autonomia.....	37
3.2.2	Mobilidade.....	38
3.2.3	Reatividade.....	39
3.2.4	Aprendizagem.....	39
3.2.5	Comunicabilidade.....	40
3.2.6	Sociabilidade.....	40
3.2.7	Continuidade temporal.....	41
3.3	Tipologia.....	41
3.4	Arquiteturas de agentes.....	45
3.5	Sistemas multiagentes.....	54
3.5.1	Taxonomia de sistemas multiagentes.....	55
3.5.2	Organizações de agentes.....	56
3.5.3	Padrões em agentes inteligentes.....	57
3.6	Sistemas multiagentes existentes.....	62
3.6.1	RETSINA.....	62
3.6.2	JADE.....	64
3.7	Técnicas de inteligência artificial.....	65
3.7.1	Programação por demonstração.....	65
3.7.2	Mineração de textos.....	66
3.7.3	Análise estatística de texto.....	68



3.7.4 Análise de linguagem natural.....	71
3.7.5 Clustering e Categorização.....	74
3.7.6 Redes neurais.....	74
4 Descrição do modelo proposto.....	77
4.1 Arquitetura básica.....	77
4.2 Interação com o usuário.....	79
4.3 Tipos de agentes.....	81
4.4 Estrutura dos agentes.....	86
4.5 Mensagens entre agentes.....	93
5 Conclusão.....	98
5.1 Limitações.....	99
5.2 Sugestões para estudos futuros.....	100

## Índice de Figuras

Figura 1 Contexto do trabalho no processo de inteligência competitiva.....	15
Figura 2 Escolas de competitividade.....	20
Figura 3 Processo de inteligência competitiva.....	25
Figura 4 Arquitetura de agentes.....	31
Figura 5 Tipologia de Agentes.....	43
Figura 6 Tipologia de agentes adotada neste trabalho.....	45
Figura 7 Arquitetura geral de um agente.....	50
Figura 8 Taxonomia de sistemas multiagentes.....	56
Figura 9 Exemplo de uma ontologia para uma empresa simples, mostrando classes e suas subclasses, relacionamentos e instâncias (indicadas pela linha tracejada).....	61
Figura 10 Visão geral dos tipos de agentes propostos.....	78
Figura 11 Casos de uso representando a interação com o usuário.....	80
Figura 12 Proposta de estrutura do agente de interface.....	88
Figura 13 Proposta de estrutura para os agentes de busca.....	89
Figura 14 Proposta de estrutura para o agente de monitoramento.....	90
Figura 15 Proposta de estrutura para o agente de banco de dados.....	91
Figura 16 Proposta de estrutura para os agentes de filtragem.....	92
Figura 17 Algumas mensagens chave na atividade de busca na Web.....	94
Figura 18 Algumas mensagens na atividade de busca em RSS.....	95
Figura 19 Algumas mensagens na atividade de monitoramento.....	96
Figura 20 Algumas mensagens na atividade de filtragem.....	97

# 1 Introdução

A World Wide Web, ou somente Web, é formada por documentos hipertexto que podem ser lidos por qualquer computador conectado à internet e equipado com um software de interpretação, o navegador, que transforma o arquivo contendo texto e referências a imagens e outros conteúdos em um documento na tela do usuário. O seu surgimento tornou a internet mais acessível ao público em geral, e tornou mais simples a tarefa de disponibilizar conteúdo na rede mundial, pois a criação dos documentos Web é tarefa do mesmo nível de complexidade da criação de um documento em um editor de textos (W3C, 2001).

Como o passar do tempo, e o crescimento da popularização da Web, cada vez mais organizações e indivíduos passaram a publicar sites na Web, desde páginas institucionais até serviços de comércio eletrônico e relacionamento com clientes e fornecedores. Juntamente a isso, surgiram várias iniciativas de publicação de conteúdo na Web, sob a forma de catálogos, revistas, jornais e até mesmo livros eletrônicos, com acesso gratuito ou pago. Os próprios governos possuem uma grande presença na Web, com o cidadão e empresa podendo acessar publicações oficiais e serviços públicos disponibilizados na rede.

Dessa forma, existe agora uma ampla gama de informações disponíveis para acesso através da Web, e estas informações podem ser usadas para alavancar a competitividade das organizações.

Um dos problemas encontrados ao acessar estas informações é a grande quantidade de informações disponíveis, tomando muito tempo do usuário para processá-las, o que acaba acarretando em tomada de decisão usando informações incompletas ou em demora para a tomada de decisão. O uso de um sistema computacional automatizado para apoiar a tarefa de

buscar informações na internet aliviaria este problema (MARCO, 1999).

O modo como se busca informações na internet atualmente é através dos sistemas de busca tradicionais, como o Google, Yahoo, Altavista, entre outros. Nesses sistemas, as buscas são realizadas através da digitação de palavras chave e o sistema então retorna os resultados que correspondam a estas palavras. O usuário também pode utilizar operadores booleanos, como OU, E, NÃO, para melhor determinar os parâmetros da busca. Isso resulta em um número muito grande de resultados, nem sempre relacionados ao assunto que se busca, devendo ser analisados manualmente. Isso torna a busca de informações na internet uma atividade que consome muito tempo com resultados não garantidos (CENDÓN, 2001; DETERS; ADAIME, 2003).

Partindo-se dessa situação, e baseando-se no trabalho realizado por Silva (2000) e de sua proposta para um sistema de agentes inteligentes para auxiliar o processo de coleta de informações na internet, decidiu-se propor um modelo para uma arquitetura de agentes inteligentes para auxiliar na tarefa de busca e monitoramento de fontes de informação na internet.

## ***1.1 Objetivos Geral e Específicos***

O objetivo geral do trabalho é propor um modelo para uma arquitetura de agentes inteligentes para auxiliar na tarefa de coleta de informações e monitoramento de fontes de informação na internet.

Objetivos específicos:

- determinar as necessidades de um sistema de coleta e monitoramento de informações para a inteligência competitiva
- examinar tecnologias de agentes inteligentes existentes
- propor um modelo de arquitetura de um sistema multiagentes
- propor as características dos agentes do modelo
- propor as interações entre os agentes no modelo
- propor as interações entre o usuário e os agentes

## **1.2 Justificativa e Motivação**

A inteligência competitiva torna-se muito importante para a sobrevivência das empresas, em um ambiente cada vez mais competitivo. A necessidade de informações corretas e em tempo hábil para a tomada de decisões, assim como a necessidade de se manter um serviço de monitoramento constante do ambiente organizacional para evitar surpresas desagradáveis, torna o processo de inteligência competitiva de vital importância para a organização (TYSON, 1998; BERNHARDT, 2003).

Existe uma necessidade de uma ferramenta de apoio ao subprocesso de coleta de informações do processo de inteligência competitiva, conforme visto em Silva (2000). Esta ferramenta automatizaria uma parte do processo, tornando mais eficaz os esforços de inteligência competitiva, sobretudo a busca de informações na internet.

O uso de ferramentas que automatizem parte do processo de obtenção de informações é essencial para o desenvolvimento futuro da inteligência competitiva, facilitando

a tarefa de transformar grandes quantidades de dados em informações úteis (MARCO, 1999).

### **1.3 Delimitação**

O presente trabalho trata da proposta conceitual de um modelo de arquitetura para um sistema multiagentes, a ser aplicado como ferramenta de apoio ao processo de inteligência competitiva. O escopo do trabalho, no tocante ao relacionamento da ferramenta com o processo de inteligência competitiva, se delimita ao subprocesso de coleta de informações. A ferramenta deve servir de apoio à coleta de informações através da automatização do monitoramento de fontes de informação na internet e busca de novas fontes.

O foco do trabalho dentro do processo de inteligência competitiva fica no processo de coleta dentro do modelo do processo de inteligência competitiva prescrito por Bernhardt (2003), que é composto de cinco etapas: planejamento e controle; coleta; processamento e exploração; análise e produção e disseminação.

### **1.4 Natureza do trabalho e metodologia**

Este trabalho é de natureza aplicada, pois se relaciona a um problema prático, o de busca e monitoramento de fontes de informações na internet. Possui também uma abordagem

qualitativa, pois não se utiliza de dados estatísticos nem produz resultados experimentais. Seu objetivo é exploratório, pois visa propor um modelo para uma arquitetura de agentes. (SILVA; MENEZES, 2005).

A metodologia de pesquisa se baseia em uma pesquisa bibliográfica que fornece as bases teóricas do processo de inteligência competitiva, a teoria de agentes e as técnicas de análise de textos existentes. Após, parte-se para propor um modelo para uma arquitetura de agentes que permita o desenvolvimento de a uma ferramenta de apoio à inteligência competitiva.

Este trabalho faz uma síntese da teoria de agentes e técnicas de inteligência artificial e o modelo proposto por Silva (2000) para a estruturação de um modelo multiagentes.

## ***1.5 Estrutura da dissertação***

O primeiro capítulo consiste nesta introdução, que estabelece os objetivos do trabalho e delimita o seu escopo. O segundo capítulo trata da competitividade, e de como ela dá origem ao processo de Inteligência Competitiva (IC). Define, então, Inteligência Competitiva, sua importância e como a internet pode ser usada na atividade de inteligência competitiva.

O terceiro capítulo trata de agentes inteligentes, suas características, arquiteturas e introduz sistemas multiagentes. Também explora algumas tecnologias de análise e mineração de textos cuja aplicação provê as habilidades de filtragem e análise da ferramenta proposta. Este capítulo dá embasamento para a descrição posterior do sistema.

O quarto capítulo descreve o sistema multiagentes proposto, os agentes que o compõe, com suas capacidades e as interações entre eles.

No quinto capítulo, é feita a conclusão do trabalho, expondo os resultados obtidos e possibilidades futuras.



## **2 Competitividade e Inteligência Competitiva**

Este capítulo abordará brevemente a questão da competitividade, com os fatores que a influenciam. Da competitividade prossegue-se à definição de Inteligência Competitiva, em que serão abordadas algumas definições da literatura com consequente escolha de uma delas para uso no decorrer do trabalho. Após a definição, será abordada a importância do processo de Inteligência Competitiva, seu papel na organização e as vantagens esperadas de sua implantação. Com esta base, será abordada então a relação entre o processo de Inteligência Competitiva e a Internet, focando as vantagens da utilização da Internet. Para finalizar, será exposto o modelo de uma ferramenta usando agentes inteligentes proposto por Silva (2000).

### **2.1 Competitividade**

De acordo com o trabalho de Junges (2004), temos que a competitividade, ou obtenção de vantagem competitiva, pode ser analisada através de quatro visões:

- Modelo de Análise Estrutural da Indústria: este modelo encara a competitividade como advinda do modo de organização da empresa, através da estrutura, comportamento e performance, e de análise de posicionamento;
- Modelo de Recursos e Competências: defende que a vantagem competitiva é alcançada através do desenvolvimento de recursos e competências, através da estrutura da indústria;

- Modelo de Processos de Mercado: sustenta que a competitividade advém da mudança, da inovação e da concorrência, com o mecanismo de preços do mercado determinando a alocação de recursos;
- Modelo de Capacidades Dinâmicas: este modelo explica a competitividade através da inovação e modificações da empresa, trabalhando em cima dos processos de decisão e das ações empreendedoras, e suas consequências.

<i><b>Dimensões</b></i>	<i><b>Descrição</b></i>
Pessoas	Clima interno da organização, ambiente de trabalho, hábitos, escala de valores, filosofia da empresa. Reflete na motivação ou desmotivação das pessoas, estilos de liderança, modelos de gestão, canais de comunicação, competência dos colaboradores, critérios de remuneração e recompensa, crescimento na empresa).
Resultados	Expressos em termos financeiros (lucro, retorno do investimento, crescimento do patrimônio líquido); marketing (fatia de mercado, volume de vendas, clientes atingidos, cobertura geográfica, imagem dos produtos e serviços; assistência técnica pós-venda); operação (volume de produção, qualidade do produto, produtividade, máquinas e equipamentos instalados).
Inovação e flexibilidade	Grau de abertura, sensibilidade, possibilidade e interesses da organização em promover mudanças e renovação no modelo de gestão, em antecipação ou como resposta às demandas mutáveis do meio ambiente. Tais mudanças, segundo o autor, dizem respeito à “destruição criativa” de conceitos, estratégias e valores,, numa atitude de constante questionamento, que permite o desenvolvimento de novos produtos, novos serviços, novos métodos e novas formas de gerenciar.

Quadro 1 Dimensões de uma empresa de sucesso  
 Fonte: BOOG (1991 apud ABREU; ABREU, 2003)

A competitividade é o posicionamento estratégico da organização, em relação a outras empresas concorrentes, que lhe permite gozar de uma posição superior do ponto de vista de concorrência. Ela pode ser vista como advinda das relações da organização com o mercado, um foco externo, ou advinda dos mecanismos e estruturas internas da organização.

A competitividade de uma organização depende de sua estratégia de atuação, de seu posicionamento. Mas para a elaboração de uma estratégia eficaz, é necessário o entendimento do que torna a organização competitiva. Desse modo, antes de elaborar a estratégia competitiva, a organização precisa conhecer o mercado, suas tendências e preferências, assim como conhecer as tecnologias e processos disponíveis, assim como é necessário um conhecimento das estratégias adotadas por seus competidores (JUNGES, 2004).

No trabalho de Abreu e Abreu(2003), a competitividade é vista além da produtividade, produção em alta escala e a custo reduzido, e do posicionamento da organização no mercado. Competitividade é visto como competição mais competência, conforme Boog (1991, apud ABREU; ABREU, 2003), resultando em que uma organização de sucesso precisa saber balancear entre as dimensões de pessoas, resultados, inovação e flexibilidade, conforme pode ser visto no quadro 1.

1. Análise de Posicionamento Foco: Indústria	2. Processos de Mercado Foco: Inovação
3. Recursos e Competências Foco: Competências Organizacionais	4. Capacidades Dinâmicas Foco: Processos

Figura 1 Escolas de competitividade.  
Fonte: Abreu e Abreu, 2003

De Vasconcelos e Cyrino, apud Abreu e Abreu (2003), tiramos a existência de quatro escolas de competitividade, ilustradas na figura 1:

- análise estrutural da indústria, que compreende a definição do cenário de

concorrentes;

- processos de mercado, que compreende a dinâmica das empresas, dos mercados e da concorrência, com foco na mudança e inovação, ao invés de na estrutura do setor produtivo; com a competitividade sendo dada por fatores externos à organização;
- recursos e competências, que explica a competitividade através das capacidades únicas da organização;
- capacidades dinâmicas, que trata da capacidade da organização de lidar com mudanças no ambiente e incertezas, através dos processos de decisão e as ações realizadas, com suas conseqüências em nível gerencial e de recursos para a organização; a competitividade sendo determinada por fatores internos à organização.

Para a criação de uma estratégia adequada à organização e ao mercado em que ela atua, é necessário conhecimento sobre o meio ambiente no qual a organização se insere, as tecnologias de base para a atuação e diferenciação da organização no mercado, assim como informações sobre a própria organização, seu potencial em termos de recursos e capacidade de comunicar e transmitir conhecimento.

O ambiente externo à organização está sempre mudando, dando origem a oportunidades e ameaças. A empresa competitiva vai ser aquela que é mais flexível, com maior capacidade de mudar para se adequar a novas situações, escapando aos perigos e tirando o máximo das oportunidades.

## **2.2 Definição de Inteligência Competitiva**

Na busca por definições do que é o processo de Inteligência Competitiva, encontra-se uma convergência de vários autores para um conceito comum, com poucas variações.

Inteligência Competitiva, IC, é um processo contínuo, com o foco no ambiente externo à organização, visando a obtenção de informações que afetem a competitividade da organização. Além da aquisição das informações, o processo também envolve a análise das mesmas e a sua transformação em conhecimento estratégico (TYSON, 1998).

Inteligência Competitiva é um processo para a aquisição de informações estratégicas, que provêm do ambiente competitivo. Ela é parte do processo de Inteligência Organizacional (SILVA, 2000).

Inteligência Competitiva é um processo, que melhora a competitividade de organizações através de um maior conhecimento dos competidores e do ambiente competitivo. É um processo contínuo, que envolve a coleta legal e ética de informações, assim como a análise das mesmas para produzir inteligência com a qual a gerência da organização pode atuar (SOCIETY OF COMPETITIVE INTELLIGENCE PROFESSIONALS, 2004).

Como inteligência, entende-se informação com relevância para as necessidades da organização, que possa ser usada para resolver problemas ou criar oportunidades.

Neste trabalho, a definição a ser usada será a de que Inteligência Competitiva é um processo contínuo de coleta e análise de informações, com o foco estratégico e no ambiente externo da organização.

## **2.3 Importância da Inteligência Competitiva**

Na literatura, encontra-se um consenso quanto ao grau de importância da Inteligência Competitiva, com os autores divergindo quanto à ênfase dada aos diferentes enfoques e produtos do processo de Inteligência Competitiva.

A importância do processo de IC advém do fato de as informações obtidas pelo processo permitirem à organização evitar ser surpreendida pelos acontecimentos, manter a sua presença no mercado, identificar oportunidades e minimizar ameaças (TYSON, 1998).

Um processo formalizado de IC tem um papel essencial no processo de pensamento estratégico nas organizações, assim como na implementação do planejamento. É uma necessidade básica de uma cultura de gerenciamento estratégico (TYSON, 1998).

As organizações encontram-se em um cenário no qual a atitude competitiva é essencial. Com o conceito de que competitividade é “manter-se 'alerta', sempre atualizado”, e que informação é um recurso estratégico para a organização, o processo de IC torna-se essencial por suprir essa necessidade de informação atualizada (SILVA, 2000).

O processo de inteligência, especialmente inteligência estratégica, é fundamental para a sobrevivência da empresa. Sem as informações obtidas pelo serviço de inteligência, o processo de planejamento estratégico não é mais do que adivinhação. Inteligência estratégica, e contra-inteligência, representam a "primeira linha de defesa" da organização, contra ameaças aos seus planos, operações e objetivos estratégicos (BERNHARDT, 2003).

Benefícios da inteligência estratégica: (BERNHARDT, 2003)

- evita surpresas estratégicas
- permite que executivos desafiem a ortodoxia facilitando a visualização de pontos cegos da

visão de negócios

- ajuda a garantir que decisões sejam baseadas em previsões e deduções ao invés de somente em sensações e experiências
- provê a gerência com uma visão única e desinteressada
- reforça a cultura competitiva na organização através do aumento da percepção dos competidores
- promove a percepção das ameaças ao capital intelectual da organização e a necessidade de contra-inteligência e contramedidas
- ajuda a diminuir a incerteza

A inteligência não é dissociada do processo estratégico, deve ser uma parte completamente integrada a ele, e deve ser institucionalizada para adicionar valor real à tomada de decisões (BERNHARDT, 2003).

O trabalho de Inteligência Competitiva sempre foi executado, se não formalmente de forma intuitiva, por qualquer empreendedor. É um fator primário para o sucesso, já que envolve saber o que querem os clientes e o que estão fazendo os concorrentes. O conhecimento, principalmente conhecimento prévio de fatos e situações importantes significa dinheiro e poder. A Inteligência Competitiva resultará na reunião e análise de informações para auxiliar o processo de tomada de decisões. As atividades de IC podem ser identificadas em quase todas as áreas de atuação humana. Quanto mais uma organização se encontra na Era do Conhecimento, mais ela precisa da IC para sobreviver e crescer no mercado (CARDOSO, 2003).

## 2.4 Processo de inteligência competitiva

Nesta seção, será apresentado o processo de inteligência competitiva com suas etapas. Esta apresentação servirá como base para o desenvolvimento da proposta do sistema.



Figura 2 Processo de inteligência competitiva.  
Fonte: Bernhardt, 2003

O processo de inteligência competitiva é uma atividade contínua, cíclica, que deve constantemente buscar informações e entregar relatórios aos membros da organização que tomam decisões (TYSON, 1998). O processo de inteligência competitiva é inseparável do processo de planejamento estratégico, e do processo de tomada de decisão estratégico da organização. Como tal, deve continuamente buscar dados e informações que auxiliem na



tomada de decisão, e deve ser subordinado ao nível estratégico da empresa (BERNHARDT, 2003).

O processo de inteligência competitiva é dividido em cinco etapas, que podem ser vistas esquematizadas na figura 4 (BERNHARDT, 2003):

- planejamento e controle: determinação das necessidades de informação dos tomadores de decisão;
- coleta: obtenção de dados e informações de todas as fontes possíveis, a serem usados na geração dos relatórios de inteligência;
- processamento e exploração: transformação de dados brutos em formas utilizáveis para a análise e geração de inteligência, como por exemplo a tradução de documentos;
- análise e produção: a avaliação e síntese das informações disponíveis e a preparação de relatórios;
- disseminação: a entrega de produtos de inteligência (relatórios) aos usuários finais.

O planejamento e controle é onde os gerentes e planejadores da organização estabelecem suas necessidades de informações e análises sobre competidores, mercados, governos e ambiente externo (BERNHARDT, 2003). Nesta etapa, o processo de inteligência competitiva deve partir dos fatores críticos de sucesso da organização e deles derivar as necessidades de informação (TYSON, 1998).

Os profissionais de inteligência devem determinar, baseadas no planejamento e requisitos obtidos na etapa anterior, as fontes de informação, a metodologia de coleta e a frequência de coleta de informações (BERNHARDT, 2003). Fontes de informação podem ser tanto informação publicada, que responde por grande parte das necessidades de informação de uma organização, como pessoas com conhecimento sobre os assuntos pesquisados,

contactadas através de uma rede de colaboradores (TYSON, 1998).

As fontes de informação são então agrupadas, de acordo com a área de interesse da organização, em um mapa de informação estratégica, usado para relacionar os requisitos de informação com as fontes que devem ser monitoradas para satisfazer estes requisitos (SILVA, 2000).

O processamento e exploração se ocupará de agrupar informações correlacionadas, tradução e interpretação de documentos, determinação de confiabilidade da fonte de informação e resolver inconsistências (BERNHARDT, 2003). É tarefa do profissional de inteligência agrupar dados correlacionados, transformando longas listas de dados numéricos em gráficos e diagramas próprios para serem analisados (TYSON, 1998).

O processo de análise e produção dos relatórios deve focar em informar alternativas e riscos, sem tentar estabelecer ações a serem tomadas pelos decisores, ao mesmo tempo em que alerta sobre possíveis problemas a longo prazo e fornece informações para as decisões que devem ser tomadas a curto prazo. Inteligência incompleta no momento da tomada de decisão pelos gestores estratégicos é muito mais proveitosa que inteligência completa após o fato ocorrido (BERNHARDT, 2003). Tomadores de decisão normalmente não possuem tempo para ler e absorver grandes quantidades de dados. Os relatórios de inteligência deve focar em ameaças e oportunidades e tendências, assim como informações chaves sobre o ambiente externo, com implicações sobre a estratégia e a sobrevivência da empresa; eles devem ser resumidos e fazer o maior uso possível de gráficos e diagramas (TYSON, 1998).

A disseminação é quando o resultado final dos esforços de inteligência é apresentado aos usuários de informação, que determinaram as necessidades informacionais a serem preenchidas. Este é o elo mais fraco do processo de inteligência competitiva. O produto final do processo de inteligência deve ser formatado de acordo com a preferência individual do usuário de informação, para facilitar a absorção do conteúdo (BERNHARDT, 2003). O tipo

de relatório e o nível de detalhe necessário varia de acordo com a função do usuário da informação. Quanto mais elevada a posição do usuário, mais estratégica sua decisão, tanto mais análise deve estar presente no relatório e este deve conter informações mais ligadas a grandes efeitos estratégicos. Para usuários de nível intermediário, mais ligados às decisões do dia-a-dia da organização, um relatório com mais quantidade de informação fatural, com menos análises de tendências e impactos se torna mais apropriado (TYSON, 1998).

## **2.5 Inteligência competitiva e a internet**

Uma grande quantidade de informação pode ser obtida de fontes de acesso público. O problema é focalizar os esforços de IC nas informações importantes no universo de informação disponível (TYSON, 1998).

Vários tipos de informação podem ser encontrados em fontes públicas. As fontes de informação pública mais comuns são listagens (*directories*) – de empresas, associações, periódicos, analistas de investimento e pessoas (executivos) – serviços de clipagem, materiais promocionais de empresas, publicações financeiras de companhias que precisam publicar as suas contas, órgãos reguladores, informações de fontes governamentais e bases de dados online (TYSON, 1998).

Com o desenvolvimento dos computadores e das redes de comunicação, o acesso à informação ficou mais fácil e barato. A Internet facilitou o acesso à informação pelo usuário final, diminuindo a dependência de profissionais especializados e consultar fontes de informação. Porém, isso gerou um problema de sobrecarga informacional, com o qual as

peessoas não conseguem assimilar as informações disponíveis e transformá-las em inteligência. Desse modo, surge a necessidade de um sistema que automatize parte do processo. Nesse contexto existem as técnicas de apresentação dos dados e as técnicas de análise automática das informações, incluindo aí análise lingüística, análise estatística, análise de linguagem natural, redes neurais, mineração de textos, clustering e classificação (MARCO, 1999).

A internet, como porta de acesso a empresas e organizações, permite encontrar grande quantidade de informação que anteriormente seria de difícil acesso, ou muito cara, tanto em recursos financeiros quanto em tempo de busca. A internet não é a única fonte de IC, mas congrega muitas fontes de informação valiosa a baixo custo (SILVA, 2000).

Assim como informações publicadas na rede, a internet permite o contato fácil com pessoas, através de e-mail ou outra ferramenta de comunicação. Pessoas a serem contatadas podem ser especialistas ou membros de outras organizações – e pessoas são uma importante fonte de IC (SILVA, 2000).

A recuperação de informações na internet esbarra no problema de que essas informações estão espalhadas em uma grande quantidade de sites, alguns deles desconhecidos. As ferramentas de busca tendem a retornar uma grande quantidade de resultados não relacionados às necessidades de informação do usuário, e o modo de se obter informação estratégica passa pelo ser humano pessoalmente acessando sites e analisando a informação. Isso requer tempo da parte do usuário (SILVA, 2000).

Outro problema encontrado no uso das ferramentas de busca tradicionais é que elas possuem pouca abrangência ou então pouca precisão, o que torna a tarefa do profissional de inteligência competitiva mais difícil (DETERS; ADAIME, 2003).

Abrangência, também chamada de revocação, e *recall* em inglês, é a medida do quanto dos documentos disponíveis e correspondentes aos critérios da busca foram recuperados (CARDOSO, 2000; EBECKEN; LOPES; COSTA, 2003).

$$\text{Precisão} = \frac{(\text{número de itens relevantes recuperados})}{(\text{número total de itens recuperados})}$$

(EBECKEN; LOPES; COSTA, 2003)

Precisão, *precision* em inglês, é a medida do quanto dos documentos recuperados realmente correspondem aos critérios de busca (CARDOSO, 2000; EBECKEN; LOPES; COSTA, 2003).

$$\text{Abrangência} = \frac{(\text{número de itens relevantes recuperados})}{(\text{número de itens relevantes na coleção})}$$

(EBECKEN; LOPES; COSTA, 2003)

Itens relevantes, nas definições acima, são os documentos que correspondem ao critério de busca usado.

Existe uma relação inversa ente precisão e abrangência, pois para melhorar a abrangência, mais documentos devem ser analisados e retornados, o que resulta em uma diminuição da precisão pela inclusão de documentos que não correspondam ao critério da busca (CHAKRABARTI, 2003).

Buscando melhorar os resultados das buscas foi instituída a *tag* “meta” nas páginas Web. Uma *tag* é uma instrução de formatação na linguagem HTML (*HyperText Markup Language* – Linguagem de Formatação de Hipertexto), usada para compor páginas Web. A *tag* “meta” permite que o criador da página insira meta-informações sobre a página, como palavras-chave, descrições, entre outras informações, que podem ser usadas por sistemas de busca. Essas meta-informações podem ser usadas para melhor situar o assunto a que a página se refere através de uma descrição ou prover uma lista de palavras-chave (W3SCHOOLS, 2005).

Outro esforço no sentido de melhorar a capacidade dos sistemas de busca de encontrar resultados que sejam mais relacionados às necessidades do usuário é a Web

Semântica (*Semantic Web*, em inglês). A Web Semântica é uma iniciativa do Consórcio da World Wide Web, o órgão que determina os padrões a serem seguidos pela Web, e parceiros em universidades e empresas ([www.w3.org/2001/sw](http://www.w3.org/2001/sw), 2005).

A Web Semântica busca desenvolver uma série de padrões que permitam que dados sejam compreendidos por computadores e seres humanos. A atividade da Web Semântica desenvolve especificações para tecnologias. As principais tecnologias centram-se em especificações de uma linguagem de descrição de conteúdos e uma linguagem de criação de ontologias. Estes padrões permitem que o criador da página monte documentos que sejam interpretados tanto por um navegador Web quanto por sistemas automatizados ([www.w3.org/2001/sw/Activity](http://www.w3.org/2001/sw/Activity), 2005).

Outra tecnologia surgindo para permitir um melhor acesso de sistemas automatizados à informação é a especificação dos Web Services. Web Services são padrões que permitem que sistemas computacionais diferentes, funcionando em plataformas diversas, trabalhem entre si, através da disponibilidade de serviços usando uma linguagem de comunicação comum, baseada no padrão XML (Extensible Markup Language). As especificações para Web Services são desenvolvidas pelo Consórcio World Wide Web ([www.w3.org/2002/ws/activity](http://www.w3.org/2002/ws/activity), 2005).

## **2.6 Inteligência competitiva, agentes e Internet**

Em Silva, 2000, foi apresentado um modelo de um sistema de agentes para usar a Internet como ferramenta de Inteligência Competitiva. As principais funções deste sistema

seriam liberar o usuário da tarefa tediosa e repetitiva de monitorar sites de informações na Internet, automatizando a tarefa e informando o usuário sempre que os sites fossem atualizados. Também deveria possuir uma capacidade básica de análise do conteúdo do site, para informar ao usuário se a atualização é relevante. Este modelo propunha uma arquitetura, que pode ser vista na Figura 3. Esta arquitetura previa quatro tipos de agentes, agentes de interface, agentes de busca, agentes de filtragem e agentes de monitoramento.

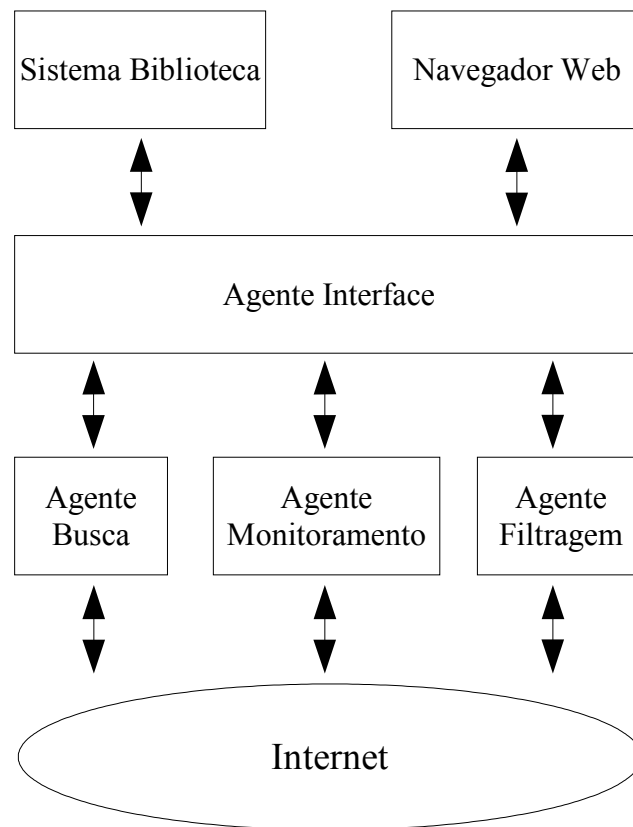


Figura 3 Arquitetura de agentes.  
Fonte: Silva, 2000.

As tarefas dos agentes de interface seriam: manter um perfil das necessidades de informação do usuário; automatizar as tarefas de busca, filtragem e monitoramento, através do controle dos outros agentes; prover uma interface gráfica para o usuário; enviar e-mails avisando o usuário de atualizações nos sites monitorados.

Os agentes de busca teriam como tarefas buscar informações, com a capacidade de identificar a relevância do conteúdo dos sites encontrados; ser autônomo, realizando filtragens e aplicando inferências; possuir capacidade de “data mining” para encontrar informações em documentos não estruturados; buscar em sites cadastrados e “varrer” a rede em busca de sites do interesse do usuário.

Aos agentes de monitoramento deveriam monitorar os sites cadastrados procurando por modificações; notificar as modificações ao agente de interface. O cadastramento das páginas a ser monitoradas poderia ser feito pelo usuário ou pelo agentes de interface, automaticamente ou pendendo aprovação do usuário.

As tarefas dos agentes de filtragem seriam manter um perfil de interesse do usuário; avaliar as páginas notificadas pelos agentes de monitoramento quanto à sua adequação ao interesse do usuário.

Analisando o modelo proposto, podemos extrair os requisitos do sistema, as tarefas que o sistema deve cumprir. Esses requisitos seriam: cadastramento dos sites, manual ou automático; manter um perfil de interesse; ter capacidade de extrair informações de documentos de texto, também chamado “text mining”; notificar o usuário através de e-mail; possuir uma interface gráfica; buscar sites de interesse na rede.



## **3 Agentes inteligentes e mineração de textos**

Este capítulo explora a tecnologia de agentes inteligentes, sua definição e suas características. Quando procura-se na literatura uma definição para agentes inteligentes, nota-se que ainda não existe um consenso quanto ao quê vem a ser exatamente um agente. Algumas definições serão exploradas e uma que seja mais adequada ao presente trabalho adotada. As propriedades dos agentes é um outro campo no qual não há muita coesão, com autores destacando características diferentes de agentes. Uma razão disso seria o fato de o campo de trabalho com agentes inteligentes ser muito vasto, com os diversos autores adotando visões mais adequadas à sua área específica.

Após a definição do que é um agente e quais suas características, uma breve visão de algumas arquiteturas de agentes será mostrada, seguida de uma exploração de sistemas multiagentes.

### **3.1 Definição**

Segundo Wooldridge (1999), não existe uma definição universalmente aceita do termo agente, mas existe um consenso de que autonomia é uma característica essencial. Um agente é um sistema que existe em um ambiente e que é capaz de atuar autonomamente sobre este ambiente para alcançar seus objetivos. Um agente é um sistema que é capaz de autonomamente sentir o seu ambiente e possui um repertório de ações para agir sobre o

ambiente e alcançar os seus objetivos.

Um agente é uma entidade que possui autonomia, não só para agir independente de outros agentes e humanos, mas para decidir como realizar a ação. Possui objetivos próprios e a capacidade de perseguí-los (GARCIA, SICHMAN, 2003).

Agentes são entidades computacionais que percebem tarefas e condições e podem decidir como agir baseado nessas condições e executar comportamentos em seu ambiente (LIU, 2001).

“Um agente autônomo é um sistema situado dentro e uma parte de um ambiente e que percebe este ambiente e age sobre o mesmo, ao longo do tempo, perseguindo sua própria agenda e para modificar o que ele perceberá no futuro” (FRANKLIN, GRAESSER, 1996, p. 4). Nessa definição, são consideradas chave a distinção entre o agente e o ambiente no qual se encontra, e a habilidade de analisar o ambiente e agir sobre o mesmo, tanto para satisfazer os seus objetivos quanto para moldar o ambiente.

Agentes são sistemas que apresentam as propriedades de autonomia, habilidade social, reatividade e pró-atividade(WOOLDRIDGE E JENNINGS, 1994).

Agente é uma entidade de software autônoma e que funciona continuamente em um determinado ambiente, interagindo com outros objetos que fazem parte deste ambiente (SHOHAM, 1997, apud THIRY, 1999).

Um agente é uma entidade autônoma, capaz de responder ao seu ambiente, comunicar-se e aprender como melhor alcançar suas metas (THIRY, 1999).

Um agente é algo que pode ser definido como capaz de perceber seu ambiente, através de algum tipo de sensor, e de agir sobre o ambiente, através de algum tipo de atuador. Um agente racional é um agente que age de forma a obter o sucesso de sua tarefa (RUSSEL, NORVIG, 1995).

Partindo das definições anteriores, será adotada neste trabalho a definição de agentes

como sistemas que sejam autônomos, e possuam capacidade de interagir com o seu ambiente, inclusive comunicar-se com outros agentes ou sistemas que componham este ambiente, e que persigam objetivos próprios, para realizar alguma tarefa ou atividade em benefício de um usuário.

### **3.2 Propriedades**

A definição de agentes dada acima, assim como várias outras na literatura, é bem ampla, podendo abarcar uma quantidade muito grande de sistemas. De modo a melhor definir e entender agentes inteligentes, uma série de propriedades, ou características, são analisadas. Estas propriedades são também usadas para a construção de uma classificação dos diversos tipos possíveis de agentes. Abaixo, um quadro lista as propriedades dos agentes, como vistas por diversos autores, e após o quadro serão discutidas as propriedades mais relevantes com mais detalhes.

A seguir segue um quadro (Quadro 2, Kern (1998) apud Oliveira, (2000)), com a uma listagem das propriedades dos agentes e os autores que as mencionam.

<b>Propriedade</b>	<b>Descrição</b>	<b>Observação</b>	<b>Referências</b>
Aprendizagem	Acumulação de conhecimento pelo agente, que adapta o seu comportamento às novas situações.	Aprendizado pode ser: <ul style="list-style-type: none"> <li>• unidirecional;</li> <li>• interativo.</li> </ul> Também referenciada como adaptatividade.	<ul style="list-style-type: none"> <li>• Wooldridge e Jennings (1996)</li> <li>• Belgrave (1995)</li> <li>• Auer (1995)</li> <li>• Lenmom apud Souza (1996)</li> </ul>
Autonomia	O agente pode ser apto a realizar ações preemptivas e independentes, que eventualmente poderão beneficiar o usuário.	Também referenciada como independência.	<ul style="list-style-type: none"> <li>• Auer (1995)</li> <li>• Wooldridge e Jennings (1996)</li> <li>• Foner (1996)</li> </ul>
Comunicabilidade	Capacidade do agente comunicar-se com outros agentes, pessoas, objetos e seu ambiente, a fim de completar suas tarefas e auxiliar os demais.	Também referenciada como habilidade social.	<ul style="list-style-type: none"> <li>• Wooldridge e Jennings (1996)</li> <li>• Franklin e Graesser (1996)</li> <li>• Belgrave (1995)</li> <li>• Auer (1995)</li> </ul>
Confiabilidade	Agentes devem servir às necessidades do usuário de maneira segura, com veracidade e benevolência, não apresentando informações falsas e objetivos conflitantes.		<ul style="list-style-type: none"> <li>• Wooldridge e Jennings (1996)</li> <li>• Auer (1995)</li> <li>• Sycara et al (1996)</li> </ul>
Cooperatividade	Capacidade dos agentes trabalharem em conjunto para concluírem tarefas mutuamente benéficas e complexas.	Também referenciada como Colaboração.	<ul style="list-style-type: none"> <li>• Sycara et al (1996)</li> <li>• Gilbert apud Souza (1996)</li> </ul>
Degradação Gradual	Capacidade do agente executar parte de uma tarefa quando existe incompatibilidade na comunicação ou domínio.		<ul style="list-style-type: none"> <li>• Foner (1996)</li> </ul>
Discurso	Troca de informações entre usuário e agente, propiciando um melhor conhecimento de suas habilidades e intenções.		<ul style="list-style-type: none"> <li>• Foner (1996)</li> </ul>
Flexibilidade	Habilidade do agente em escolher dinamicamente as ações e a sequência de execução das mesmas.		<ul style="list-style-type: none"> <li>• Jennings (1995)</li> <li>• Auer (1995)</li> <li>• Franklin e Graesser (1996)</li> </ul>
Inteligência	Conjunto de recursos, atributos e características as quais abilitam um agente a decidir quais ações tomar.	O raciocínio desenvolve-se em três cenários: <ul style="list-style-type: none"> <li>• regras;</li> <li>• conhecimento;</li> <li>• evolução artificial.</li> </ul>	<ul style="list-style-type: none"> <li>• Auer (1995)</li> <li>• Gilbert e Janca (1996)</li> <li>• Nissen (1995)</li> </ul>

<b>Propriedade</b>	<b>Descrição</b>	<b>Observação</b>	<b>Referências</b>
Mobilidade	Habilidade do agente em mover-se de uma máquina para outra, entre diferentes plataformas e sistemas, preservando seu estado interno.	Apresenta três graus: <ul style="list-style-type: none"> <li>• estático;</li> <li>• roteiros móveis;</li> <li>• móveis com estado.</li> </ul>	<ul style="list-style-type: none"> <li>• Auer (1995)</li> <li>• Belgrave (1995)</li> <li>• Ochar (1996)</li> </ul>
Persistência	Capacidade de manter um estado interno conciso através do tempo, o qual não é modificado ao acaso.	Também referenciado como continuidade temporal.	<ul style="list-style-type: none"> <li>• Belgrave (1995)</li> <li>• Etzioni e Weld (1995)</li> </ul>
Personalização	O usuário determina como o agente deve interagir, este aprende as características, preferências e hábitos de seus usuários.		<ul style="list-style-type: none"> <li>• Gilbert e Janca (1996)</li> <li>• Kearney (1996)</li> </ul>
Planejamento	Habilidade de sintetizar e escolher entre diferentes sequências de ações pretendidas para alcançar seus objetivos.		<ul style="list-style-type: none"> <li>• Belgrave (1995)</li> <li>• Auer (1995)</li> </ul>
Pró-atividade	Agentes são capazes de exibir comportamento oportunístico e orientado a objetos, tomando a iniciativa quando achar apropriado.		<ul style="list-style-type: none"> <li>• Wooldridge e Jennings (1996)</li> <li>• Belgrave (1995)</li> </ul>
Reatividade	Capacidade de perceber e responder às modificações que ocorrem no seu ambiente.	Também referenciado como "sentir e agir".	<ul style="list-style-type: none"> <li>• Hermans (1996)</li> <li>• Franklin e Graesser (1996)</li> </ul>
Representabilidade	Representação do usuário em termos de suas ações e não mera existência.	Pode ser de dois modos: <ul style="list-style-type: none"> <li>• passiva;</li> <li>• ativa.</li> </ul>	<ul style="list-style-type: none"> <li>• Wooldridge e Jennings (1995)</li> <li>• Auer (1995)</li> </ul>
Responsabilidade	O agente requisitado a realizar uma tarefa, é responsável pela sua conclusão.		<ul style="list-style-type: none"> <li>• Auer (1995)</li> </ul>

Quadro 2 Atributos mais relevantes de agentes.

Fonte: Kern (1998), apud Oliveira (2000)

Algumas propriedades são mais significativas, pois são propriedades que definem uma entidade como agente, ou delimitam melhor as capacidades do agente. Como as propriedades são usadas como um modo de se classificar e organizar os diversos tipos de agentes, algumas propriedades são mais importantes para o presente trabalho por serem propriedades importantes para o funcionamento do sistema discutido. A seguir serão comentadas as propriedades dos agentes consideradas mais significativas para este trabalho.

### **3.2.1 Autonomia**

Autonomia é a propriedade do agente de agir sem intervenção externa e buscar seus objetivos (THIRY, 1999).

Um agente possui autonomia de execução, que é a capacidade de atuar sem a interferência de outros agentes ou usuário. Também possui autonomia de decisão, que é a capacidade de decidir a ação a ser tomada baseado somente nos seus conhecimentos prévios, na sua percepção do ambiente e nos seus objetivos (GARCIA, SICHMAN, 2003).

Um agente tem um comportamento individual e age em resposta à outros agentes ou ao ambiente em que existe de forma independente. Para isso, o agente deve possuir uma base de conhecimento própria que estabelece seu comportamento (SPERB, 2002).

A autonomia é uma propriedade que define o agente, sem ela estamos falando de um programa de computador, não um agente (FRANKLIN, GRAESSER, 1996).

### **3.2.2 Mobilidade**

A mobilidade é a propriedade de um agente que pode ser transmitido para outros lugares, realizar suas tarefas e depois retornar. Isso acarreta em maior complexidade na implementação dos agentes, devido à necessidade de se adaptar a várias plataformas e sistemas (MARTINS, 2002).

Mobilidade é a propriedade de um agente de mover-se e ser executado em outros sistemas (GARCIA, SICHMAN, 2003).

A capacidade de mover-se entre sistemas de uma rede é interessante, especialmente em agentes de busca, porém pode vir a causar sobrecargas na rede. Também existe um risco de segurança associado a agentes móveis, sendo necessária a implantação de sistemas de segurança que protejam a máquina hospedeira e só permitam acesso a agentes autorizados (THIRY, 1999).

### **3.2.3 Reatividade**

Reatividade é a propriedade de um agente perceber mudanças em seu ambiente e adequar seu comportamento a essas mudanças (THIRY, 1999; MARTINS, 2002).

“Reatividade é a capacidade de reagir às mudanças do ambiente a partir do reconhecimento de um contexto conhecido” (GARCIA, SICHMAN, 2003, p.275).

A reatividade é outra das propriedades cuja existência é necessária para que o sistema seja considerado um agente. Esta propriedade determina que o agente seja capaz de responder em tempo hábil a mudanças no seu ambiente (FRANKLIN, GRAESSER, 1996).

### **3.2.4 Aprendizagem**

A aprendizagem é um dos atributos que definem um agente inteligente. Um agente que não tenha sucesso em realizar seus objetivos deve aprender com suas experiências para melhorar a sua habilidade (THIRY, 1999).

Aprendizagem é a habilidade de um agente de modificar seu comportamento baseado em suas experiências e no resultado de suas ações no ambiente (SPERB, 2002 LIU, 2001).

### **3.2.5 Comunicabilidade**

Comunicabilidade é a capacidade de agentes trocarem mensagens com outros agentes ou com outras entidades, tais como usuários. Torna-se importante quando existem mais de um agente envolvido ou o agente precisa receber instruções ou enviar resultados.

Comunicabilidade é a capacidade que um agente tem de interagir com outros agentes para atingir suas metas. Quando vários agentes atuam juntos, tem-se um sistema multi-agentes (GARCIA, SICHMAN, 2003).

Comunicabilidade é a capacidade dos agentes interagirem entre si e com usuários e ambiente, através da troca de mensagens, conhecimento, colaboração, negociação, coordenação, ou outro tipo de interação que exija comunicação (MARTINS, 2002).

A comunicabilidade, capacidade de se comunicar com outros agentes, usuário ou mesmo o ambiente, é uma propriedade importante quando mais de um agente está envolvido (THIRY, 1999).



### **3.2.6 Sociabilidade**

Sociabilidade é a capacidade de interação com outros agentes, podendo ser cooperativa ou competitiva (SPERB, 2002). Ligada à comunicabilidade (THIRY, 1999).

### **3.2.7 Continuidade temporal**

A continuidade temporal do agente, o fato de ele ser um processo que é executado continuamente, cujo fim geralmente é determinado por ele próprio, é uma das propriedades que definem um agente (FRANKLIN, GRAESSER, 1996).

A continuidade temporal é a capacidade de manter um estado interno através do tempo, incluindo os objetivos do agente (MARTINS, 2002).

## **3.3 Tipologia**

Na literatura, encontra-se uma grande variedade de classificações de agentes. A seguir citaremos algumas.

Franklin e Graesser propõem vários tipos possíveis de classificação de agentes inteligentes. Entre eles estão uma classificação baseada nas propriedades que o agente

apresenta, segundo a qual um agente poderia ser classificado como agente móvel, ou agente com aprendizado, ou agente móvel com aprendizado. Ou então uma classificação baseada nas tarefas dos agentes, segundo a qual um agente poderia ser agente de busca de informações ou agente de filtragem de e-mail. Outra classificação possível seria de acordo com o ambiente no qual o agente atua, podendo um agente ser um agente de software, outro um agente de vida artificial. Outra classificação sugerida é quanto ao tipo de estrutura de controle adotada pelo agente, ou seja, qual o princípio tecnológico da tomada de decisão pelo agente. Dessa forma, um agente pode ser algorítmico, baseado em regras, baseado em casos, em redes neurais, baseado em lógica fuzzy, entre outros (FRANKLIN, GRAESSER, 1996).

Uma outra classificação possível seria organizar os diversos agentes em uma taxonomia remanescente do método adotado para classificar os seres vivos, em forma de uma árvore, com a raiz representando um “reino” de agentes, subdivididos em agentes biológicos, robóticos e computacionais, estes últimos subdivididos em agentes de vida artificial e agentes de software, estes últimos ainda subdivididos em agentes orientados a uma tarefa específica, agentes de entretenimento e vírus; e assim por diante (FRANKLIN, GRAESSER, 1996).

Eles, Franklin e Graesser (1996), também propõe uma taxonomia baseada nas propriedades dos agentes. Cada propriedade daria origem a uma classificação binária, do tipo agente móvel ou não-móvel, agente que aprende ou sem aprendizado. Essas classificações binárias seriam combinadas em uma matriz multi-dimensional, com cada classificação binária sendo uma dimensão. Desse modo, cada célula da matriz multi-dimensional representaria uma série de propriedades, e seria uma categoria de classificação na taxonomia.

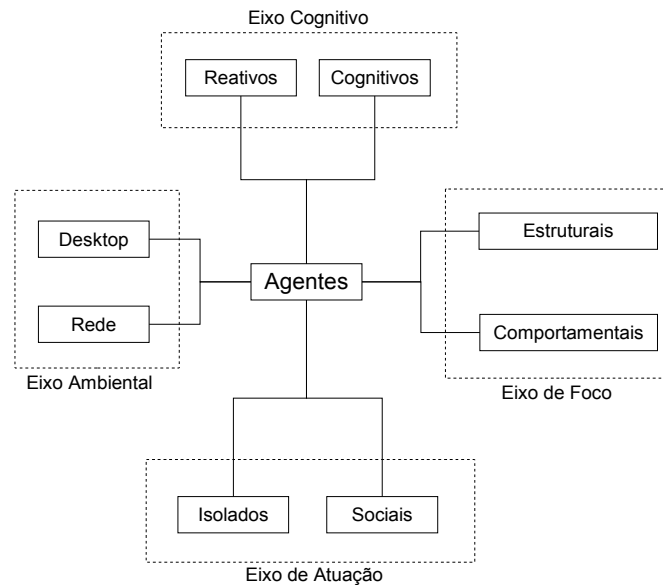


Figura 4 Tipologia de Agentes.  
 Fonte: GARCIA, SICHMAN, 2003

Uma classificação semelhante, baseada em múltiplas dimensões, é proposta por Garcia e Sichman (2003), e os classifica em quatro eixos, conforme pode ser visto na figura 4:

Os agentes são classificados segundo quatro eixos:

- eixo cognitivo: o agente pode ter uma representação interna do ambiente e um modelo racional de decisão (agente cognitivo) ou pode simplesmente reagir a estímulos externos (agente reativo);
- eixo de foco: o agente pode enfatizar semelhanças físicas com seres humanos (agente estrutural) ou enfatizar semelhanças de comportamento (agente comportamental);
- eixo de atuação: o agente pode atuar sozinho (agente isolado) ou com outros agentes (agente social);
- eixo ambiental: o agente pode atuar no desktop (agente de desktop) ou em uma rede, como a internet ou uma intranet (agente de rede).

A classificação a ser adotada neste trabalho é a de Franklin e Graesser (1996), de

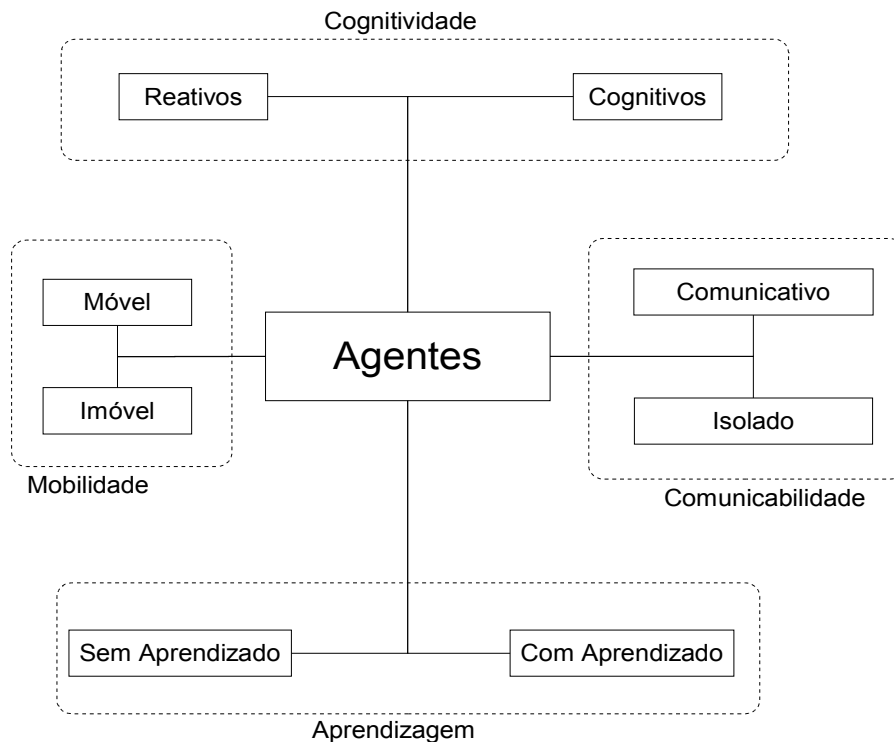


Figura 5 Tipologia de agentes adotada neste trabalho.  
Adaptado a partir de FRANKLIN, GRAESSER, 1996

classificação multidimensional, e pode ser vista na figura 5. As dimensões a serem consideradas serão:

- aprendizagem: classifica o agente de acordo com sua capacidade de aprender, de modificar suas respostas ao ambiente à medida que interage com o mesmo, ou que recebe instruções para modificá-las;
- comunicabilidade: classifica o agente de acordo com sua capacidade de se comunicar com outros agentes, trocando informações sobre suas atividades e dados que está manipulando;
- mobilidade: classifica o agente de acordo com a capacidade do agente de se deslocar de um ambiente computacional para outro, enquanto realiza suas tarefas;
- cognitividade: classifica o agente segundo sua capacidade de criar uma interpretação do ambiente à sua volta e, partindo desta representação, decidir o curso de ação que melhor atinge seus objetivos. Um agente não cognitivo é um

agente reativo.

O motivo da adoção desta classificação, é o fato de ela se adequar bem ao presente trabalho, por considerar as propriedades mais significativas para os agentes que compõem o sistema, ao mesmo tempo em que é intuitiva e fácil de visualizar.

### **3.4 Arquiteturas de agentes**

Uma arquitetura de um agente é a descrição das partes de um agente, suas estruturas de dados, as operações que podem ser desenvolvidas nas estruturas de dados e o fluxo de controle entre as estruturas de dados (WOOLDRIDGE, 1999).

Segundo Wooldrige, agentes podem ser vistos como pertencentes uma de quatro arquiteturas diferentes: agentes baseados em lógica, agentes reativos, agentes BDI (do inglês para crença- desejo-intenção) e agentes com arquitetura em camadas (WOOLDRIDGE, 1999).

Agentes baseados em lógica tem suas decisões tomadas através das técnicas de Inteligência Artificial clássicas (IA simbólica), representando o ambiente e as ações que podem ser realizadas por símbolos, com regras sintáticas definindo as manipulações que podem ser realizadas nestes símbolos. Os símbolos são fórmulas lógicas, e as manipulações a serem efetuadas são deduções lógicas, ou provas de teoremas. Nesta abordagem, o processo de decisão é visto como dedução lógica, as estratégias de decisão são modeladas como uma teoria lógica e o processo de escolha de uma ação é um problema de prova lógica. Abordagens lógicas possuem uma semântica clara e coerente. Os problemas dessa abordagem aparecem a medida que a complexidade do problema cresce, pois as regras semânticas e a transcrição do

ambiente em símbolos lógicos devem ser feitas manualmente e se tornam muito trabalhosas se complexas, as entidades do mundo real nem sempre são facilmente traduzidas em símbolos claros, e o tempo computacional para encontrar a solução de um problema complexo cresce muito (WOOLDRIDGE, 1999).

Agentes reativos são modelados a partir do princípio de que o comportamento inteligente é resultado da interação do agente com o ambiente no qual existe. Um comportamento inteligente e complexo pode emergir de um conjunto de comportamentos simples. Entre as várias arquiteturas que aplicam esta abordagem, a mais conhecida é a chamada “subsumption architecture”. Esta arquitetura define uma série de comportamentos, organizados em uma hierarquia, na qual um comportamento hierarquicamente superior, quando ativado, inibe a ativação dos outros comportamentos. Esses comportamentos consistem em reações às condições do ambiente, sendo portanto mais simples de serem modelados e implementados. Quando uma determinada condição no ambiente ativa um comportamento hierarquicamente superior, possivelmente uma reação a uma condição específica do meio, os demais comportamentos são ignorados. Se o ambiente não aciona o comportamento superior, uma ação inferior, potencialmente mais genérica, poderá então ser acionada. Através da definição dos comportamentos e de sua organização hierárquica, pode-se chegar a comportamentos complexos e adaptáveis no agente. As abordagens reativas são simples de serem modeladas e implementadas, são econômicas quanto ao tempo computacional e robustas quanto a falhas. Como os agentes não possuem um modelo do ambiente, são restritos a ações que possam ser realizadas com as informações disponíveis localmente, o que torna seu comportamento insensível ao ambiente que não podem perceber, inclusive estados anteriores do ambiente, fazendo com que sejam agentes atuantes somente a curto prazo. Devido à falta do modelo do ambiente, o agente também não pode aprender com suas experiências, sendo o seu comportamento previamente programado. O comportamento

emergente é difícil de ser previsto a partir das ações mais simples e suas interações, assim como a interação dessas ações com o ambiente. Essa dificuldade de prever o comportamento do agente leva a um processo de modelagem complexo e sem metodologia clara, e a medida que a quantidade de comportamentos individuais vai aumentando, surgem dificuldades para organizá-los em uma estrutura hierárquica que execute as tarefas necessárias de forma previsível (WOOLDRIDGE, 1999; LIU, 2001).

Agentes BDI, cujo nome vem da sigla em inglês para crença-desejo-intenção, partem do entendimento filosófico do chamado raciocínio prático, que é o processo usado por seres humanos para decidir a cada momento que ação realizar para atingir um objetivo. O raciocínio prático envolve dois processos, deliberação e raciocínio meios-fins. A deliberação é o processo que determina quais os objetivos do agente, e o raciocínio meios-fins determina como o agente atingirá estes objetivos. Um agente BDI possui sete componentes principais:

- conjunto de crenças atuais: representam as informações que o agente possui sobre seu ambiente;
- função de atualização de crenças: utiliza as crenças atuais e informações advindas dos sensores do agente para atualizar as crenças do mesmo, sua representação do ambiente;
- função geradora de opções: baseada nas intenções do agente (seus objetivos) e suas crenças (como percebe o ambiente), gera um conjunto de opções de ação;
- conjunto de opções atuais (desejos): representa as várias ações que o agente pode realizar;
- função filtro: representa o processo deliberativo do agente, com base nas crenças, desejos e intenções atuais, determina novas intenções;
- conjunto de intenções: representam o foco das atividades atuais do agente, seus objetivos;

- função seleção de ação: determina qual ação o agente realizará baseada nas suas intenções atuais.

As vantagens desta arquitetura são a geração de uma decomposição funcional bem definida para o agente, e usa um processo de decisão baseado no que os seres humanos usam no seu dia-a-dia. A principal dificuldade é a implementação destas funções de modo computacionalmente eficiente (WOOLDRIDGE, 1999).

Arquiteturas em camadas organizam o agente como um conjunto de subsistemas em camadas, cada um realizando um comportamento desejado do agente. Desse modo, o comportamento reativo pode ser realizado em uma camada, com o comportamento pró-ativo em outra, por exemplo. Os agentes em camadas podem ser divididos em horizontais e verticais, de acordo como a forma como as camadas são organizadas. Nos arranjos horizontais possuem todas as suas camadas com acesso às informações dos sensores e aos atuadores do agente, enquanto que nos arranjos verticais somente uma camada tem acesso aos sensores, e, opcionalmente, outra aos atuadores, com as demais camadas recebendo suas informações umas das outras.

Um agente com arquitetura horizontal comporta-se como se existissem vários sub-agentes opinando sobre o comportamento a ser adotado. Para evitar que o agente se comporte de forma anormal, geralmente é adicionada uma camada que atua como controladora e determina qual camada comandará o agente. A principal vantagem deste arranjo é a facilidade de se adicionar novos comportamentos simplesmente adicionando novas camadas. A desvantagem é que um grande número de camadas gera uma situação complexa de controle, com a camada de controle tornando-se um gargalo no desempenho do agente.

O arranjo vertical atua como se o agente fosse uma organização tradicional, com as camadas atuando como níveis funcionais hierárquicos. Cada camada realiza sua atividade, de acordo com sua capacidade e, encontrando uma situação que não consiga processar, envia-a



para o nível superior. Este arranjo é mais simples que o horizontal, diminuindo a complexidade das interações entre as camadas devido a um número menor de comunicações possíveis entre elas. Porém, ele é menos flexível que o arranjo horizontal, e mais vulnerável a falhas, já que a comunicação passa por todas as camadas e a falha de uma delas paralisa o processo de decisão (WOOLDRIDGE, 1999).

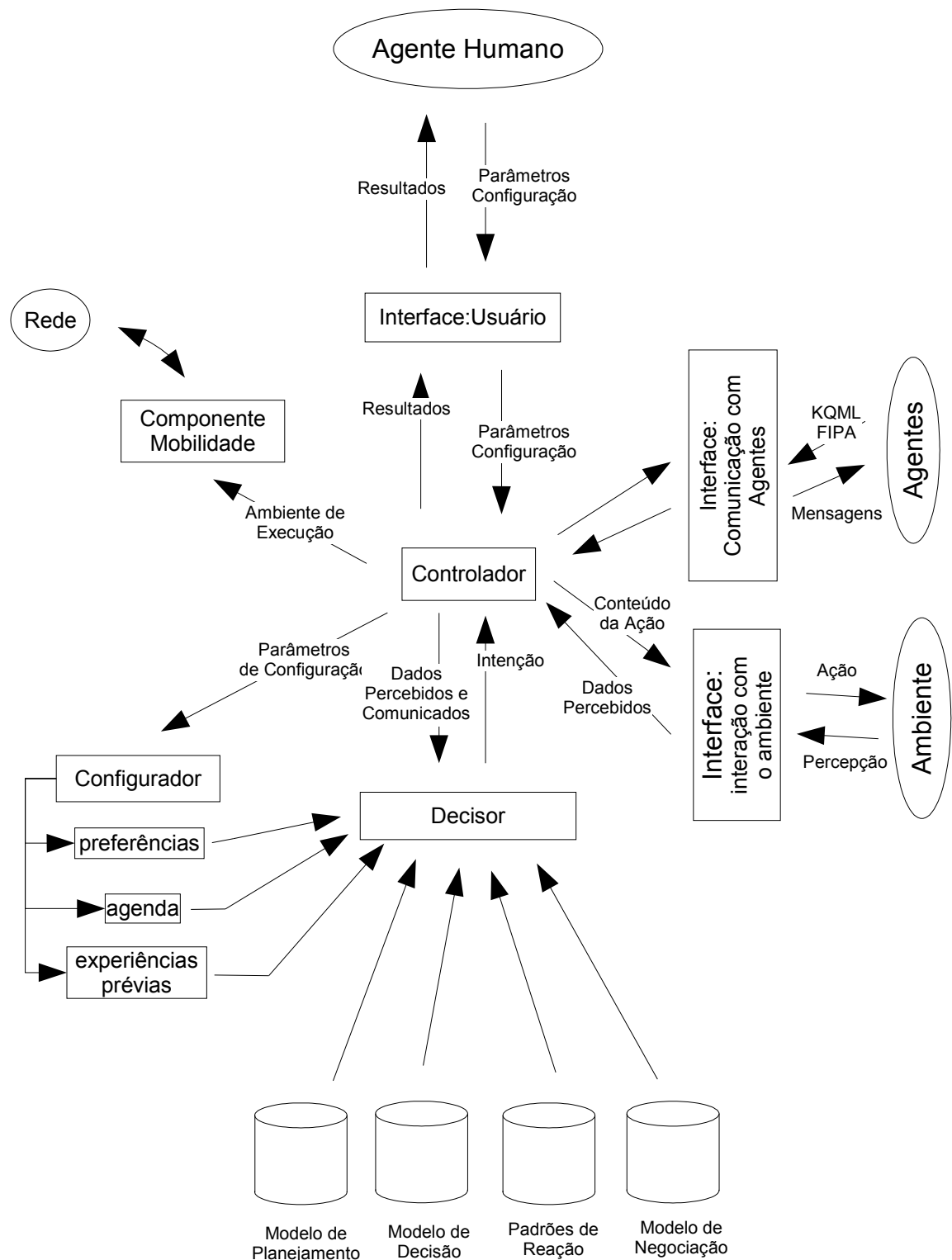


Figura 6 Arquitetura geral de um agente  
 Fonte: GARCIA, SICHMAN, 2003

Uma proposta de uma arquitetura geral, abstrata, de um agente, é mostrada na figura

6. Essa arquitetura busca mostrar os diferentes módulos necessários em um agente, com suas respectivas inter-relações. O módulo controlador é o responsável por controlar a execução dos processos internos do agente; o módulo decisor utiliza modelos de planejamento, decisão, reação e de negociação para decidir qual o melhor curso de ação do agente; o módulo configurador é responsável por atualizar os parâmetros internos do agente em resposta às necessidades do usuário, a mudanças no ambiente ou a interações com outros agentes; o módulo de interface com o usuário permite que o agente receba instruções e retorne resultados; o módulo de interface com outros agentes permite a comunicação entre agentes, seja através de objetos compartilhados, sistema de quadro-negro ou através de uma linguagem de comunicação e ontologias compartilhadas; o módulo de interface com o ambiente implementa a capacidade de sentir o ambiente no qual o agente existe e a capacidade de agir sobre o mesmo; o módulo componente de mobilidade permite aos agentes móveis se moverem entre várias plataformas e sistemas diferentes (GARCIA, SICHMAN, 2003).

Outra arquitetura é a Seleção de Ação (Action Selection), na qual o agente recebe um estímulo do ambiente, e uma série de ações possíveis previamente existentes é avaliada e uma ação é escolhida e realizada. Essa arquitetura é orientada a objetivos, relevante à situação presente, reativa, rápida, robusta e focada no futuro. O conhecimento do agente, as ações que serão analisadas, deve ser alimentado no mesmo, através de programação prévia ou algum método de aprendizagem (LIU, 2001).

Outra arquitetura é a arquitetura Motif. Esta arquitetura é multi-nível, e foi criada para criar agentes capazes de gerar e organizar comportamentos em um ambiente dinâmico e desconhecido. O componente básico desta arquitetura é o Motif, uma unidade computacional auto-organizável. Vários motifs podem ser agrupados juntos, de acordo com suas necessidades. Os vários motifs estão inter-relacionados, e trocam dados e/ou intervenções, que são instruções que controlam a troca de informações. O sistema desenvolve um

comportamento devido ao fato de que os motifs se auto-organizam e enviam sinais que causam outros motifs a se auto-organizarem. Essa cadeia de adaptações atinge um equilíbrio, no qual as mudanças nos motifs são pequenas (LIU, 2001).

O quadro 3 resume as diversas arquiteturas analisadas e suas características.

<i><b>Arquiteturas</b></i>	<i><b>Características</b></i>	<i><b>Pontos positivos</b></i>	<i><b>Pontos negativos</b></i>
Baseada em lógica	Utiliza uma abordagem de lógica simbólica para o processo de decisão	Processos lógicos de fácil entendimento, formalismo lógico bem entendido	Caro em tempo de computação, difícil de modelar comportamentos muito complexos.
Agentes reativos	Conjunto de reações ao ambiente, organizados de forma a criar comportamentos complexos a partir de reações simples	Simples de modelar, barato em termos computacionais, robusto	Comportamento emergente difícil de prever e projetar.
Agente BDI	Usa o raciocínio prático para estabelecer um processo de decisão	Modelagem intuitiva, decomposição funcional do agente	Difícil implementação eficiente das funções
Arquitetura em camadas	Agente visto como um conjunto de subsistemas em camadas	Flexível, expansível	Complexidade das interações entre agentes torna sistemas complexos computacionalmente caros
Arquitetura modular	Abstrata, determina módulos que cumprem funções distintas do agente	Especifica o fluxo de informação entre os módulos, separação entre os módulos por tarefa do agente	Alta complexidade, descrição abstrata da implementação
Arquitetura de seleção de ação	Os estímulos do ambiente são avaliados e comparados com uma série de ações para determinar a ser realizada	Orientado a objetivos, reativa, rápida execução	Comportamento deve ser especificado como conjunto de ações

<i>Arquiteturas</i>	<i>Características</i>	<i>Pontos positivos</i>	<i>Pontos negativos</i>
Motif	Modelagem multi-nível, auto-organizável, com unidades funcionalmente semelhantes	Flexível, adaptável	Auto organização difícil de prever e projetar o comportamento

Quadro 3 Comparação das arquiteturas de agentes

### **3.5 Sistemas multiagentes**

Existem duas abordagens para o uso de múltiplos agentes, a Resolução Distribuída de Problemas (RDP) e Sistemas Multiagentes (SMA). Na RDP, os agentes e a forma de organização dos mesmos são desenvolvidos a partir de um problema específico a ser solucionado. Já na abordagem SMA, os agentes existem previamente em um universo multiagentes, independentes da existência de um problema prévio. O problema a ser resolvido é então proposto à sociedade de agentes, que se organiza de forma a solucioná-lo (GARCIA, SICHMAN, 2003).

Sistemas Multiagentes (SMA) possuem as seguintes características (HUHNS, STEPHENS, 1999):

- sistemas multiagentes provêm uma infraestrutura de comunicação e interação entre os agentes;
- sistemas multiagentes são abertos e não possuem um controle centralizado;
- MAS contém agentes que são distribuídos e autônomos.

Resolução Distribuída de Problemas (Distributed Problem Solving – RDP) é uma abordagem a sistemas multiagentes que parte do pressuposto de que os agentes ou foram construídos para trabalharem juntos, ou, devido a regras do conjunto de agentes, o comportamento individualístico é desencorajado. Dessa forma, os agentes cooperam para a solução de um problema, e o domínio da RDP é a criação de um conjunto de agentes que trabalhem juntos de forma competente. Geralmente, um conjunto de agentes, tentando resolver um problema, precisa antes de tudo, desenvolver um plano. Por isso, a solução de problemas de planejamento está intimamente ligada à resolução distribuída de problemas (DURFEE, 1999).

### **3.5.1 Taxonomia de sistemas multiagentes**

Os sistemas Multiagentes podem ser classificados de acordo com as suas características em alguns eixos, como pode ser visto na figura 7 (GARCIA, SICHMAN, 2003):

- eixo de perspectiva: um sistema multiagentes pode ter uma perspectiva de simulação social ou de resolução de problemas;
- eixo de abertura: indica se um sistema pode alterar a quantidade de agentes que o compõe dinamicamente;
- eixo de granularidade: um sistema com baixa granularidade contém poucos agentes, enquanto que um sistema com alta granularidade pode conter milhares de agentes;
- eixo de composição: classifica o sistema baseado nos tipos de agentes presentes,

em um sistema homogêneo, se for composto por um único tipo de agente, ou heterogêneo, se vários tipos de agente existem no mesmo sistema;

- eixo da interação: classifica o sistema baseado nas várias interações sociais possíveis entre os agentes.

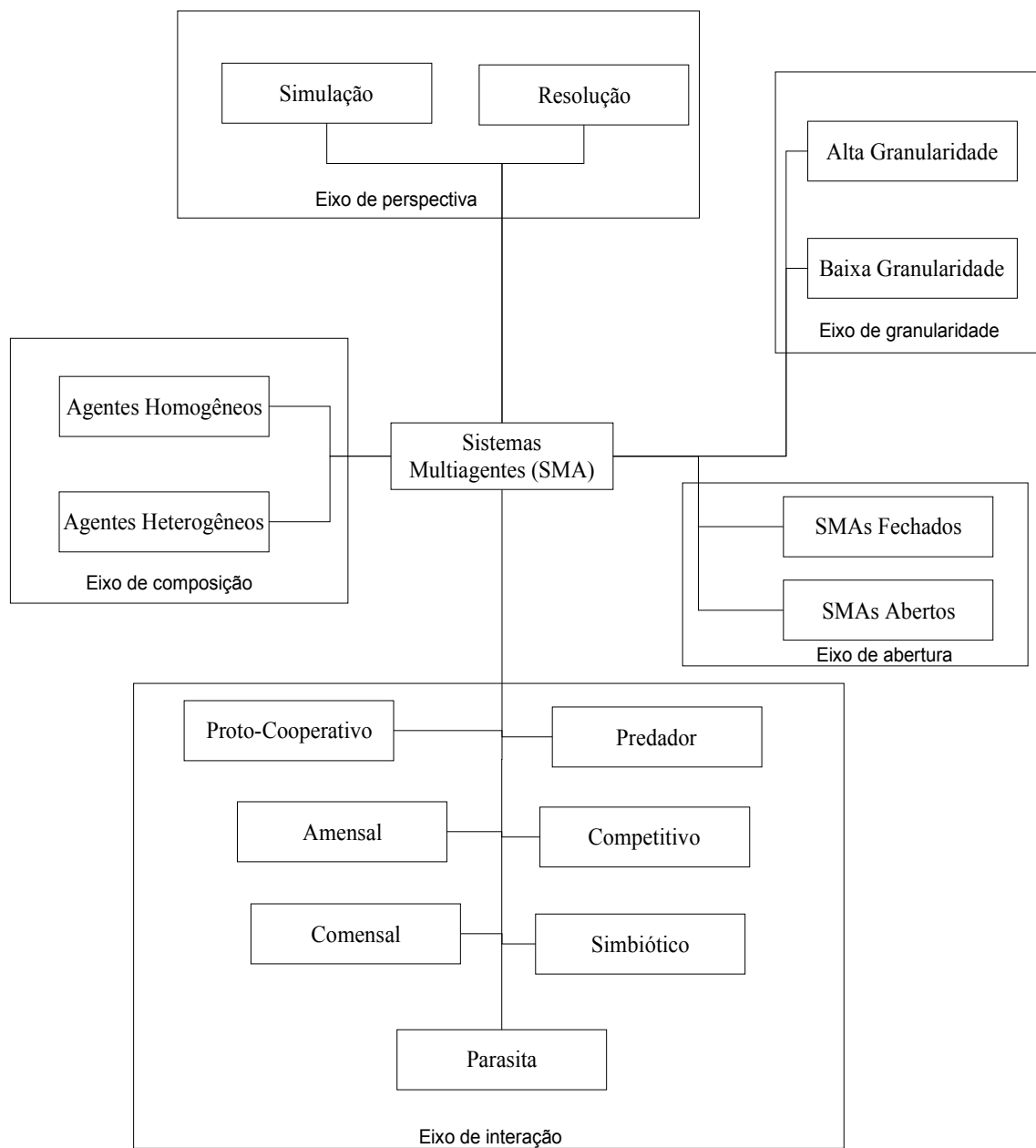


Figura 7 Taxonomia de sistemas multiagentes.  
Fonte: GARCIA, SICHMAN, 2003

### **3.5.2 Organizações de agentes**

Uma organização delimita o tipo e o formato das interações entre os agentes, diminuindo a complexidade da sociedade de agentes. Organizações podem ser modelos estáticos ou modelos dinâmicos. Nos modelos estáticos ou descendentes (top-down), a estrutura da organização é pré-definida, baseada no problema que os agentes resolverão. Nos modelos dinâmicos ou ascendentes (bottom-up), os agentes possuem seus próprios objetivos e as interações sociais surgem dinamicamente à medida que os agentes buscam satisfazê-los (GARCIA, SICHMAN, 2003).

Uma organização possui um projeto, um conjunto de procedimentos e canais de comunicação que ligam os diversos componentes da organização, sejam eles agentes, tarefas, recursos ou capacidades, e estabelece as atribuições e responsabilidades de cada membro (CARLEY E GASSER, 1999).

Modelos baseados em teoria organizacional são baseados nos estudos e conhecimentos empíricos de como as organizações humanas são formadas e operam e os princípios das teorias da organização. Os modelos de organização geralmente são multicamadas, com modelos genéricos descrevendo o comportamento de seções da organização e modelos diferentes detalhando o comportamento de agentes individuais nas seções (CARLEY E GASSER, 1999).



### 3.5.3 Padrões em agentes inteligentes

Com o desenvolvimento da tecnologia de agentes, surgiram padrões de construção de agentes para sistematizar e integrar os esforços dos diversos grupos que trabalham com agentes. Dentre estes padrões, podemos destacar o padrão de comunicação KQML e o padrão FIPA para sistemas multiagentes.

#### 3.5.3.1 Comunicação entre agentes

(KQML-performative

:sender <word>

:receiver <word>

:language <word>

:ontology <word>

:content <expression>

...)

Quadro 4 Formato de uma mensagem KQML  
Fonte: Huhns e Stephens (1999)

O protocolo KQML é uma linguagem estruturada em três camadas: comunicação, mensagem e conteúdo. A camada de comunicação estabelece informações que identifiquem o emissor, o receptor e multiplicidade do pacote de informações. A camada de mensagem estabelece qual o tipo da mensagem, se uma assertiva, uma consulta ou um comando, assim como estabelece ontologias, a linguagem do conteúdo e outras informações. A camada de

conteúdo é a razão da mensagem, podendo ser em uma linguagem de conteúdo específica diferente. A linguagem KQML trata agentes como bases de conhecimento virtuais, tendo o agente que implementar um encapsulamento que permita que outros agentes o vejam como tendo as características de uma base de conhecimento (GARCIA, SICHMAN, 2003; THIRY, 1999; FININ et al., 1993).

A linguagem KQML é formada por performativas e argumentos. A estrutura de uma mensagem KQML pode ser vista no quadro 4.

Os argumentos são palavras-chave precedidas por dois pontos. As performativas KQML são modeladas baseadas nas performativas de atos de fala, e são organizadas em sete categorias. As performativas podem ser, entre outras (HUHNS; STEPHENS, 1999):

- performativas de consultas básicas (evaluate, ask-one, ask-all, ...)
- performativas de consulta com respostas múltiplas (stream-in, stream-all, ...)
- performativas de resposta (reply, sorry, ...)
- performativas de informação genéricas (tell, achieve, cancel, untell, unachieve, ...)
- performativas de geração (standby, ready, next, rest, ...)
- performativas de definição de capacidade (advertise, subscribe, monitor, ...)
- performativas de rede (register, unregister, forward, broadcast, ...)

A linguagem KQML, provê então uma forma de comunicação que agentes possam entender. Isso garante que todos os agentes entendam que tipo de comunicação está sendo estabelecida. O conteúdo da comunicação é estabelecido através das palavras-chave : *language*, que determina a linguagem de conteúdo da comunicação; : *ontology*, que determina a ontologia usada na comunicação, seu vocabulário; : *content*, que contém o conteúdo da comunicação. Para entender a comunicação o agente deve entender a linguagem e ter acesso à ontologia da mensagem (HUHNS; STEPHENS, 1999).

Uma linguagem de conteúdo é KIF (Knowledge Interchange Format), uma linguagem

lógica de descrição de conteúdo. A KIF é uma linguagem que se baseia em lógica de predicados, e pode ser usada para representar informações sobre fatos e objetos do mundo real, assim como seqüências de atividades (HUHNS; STEPHENS, 1999).

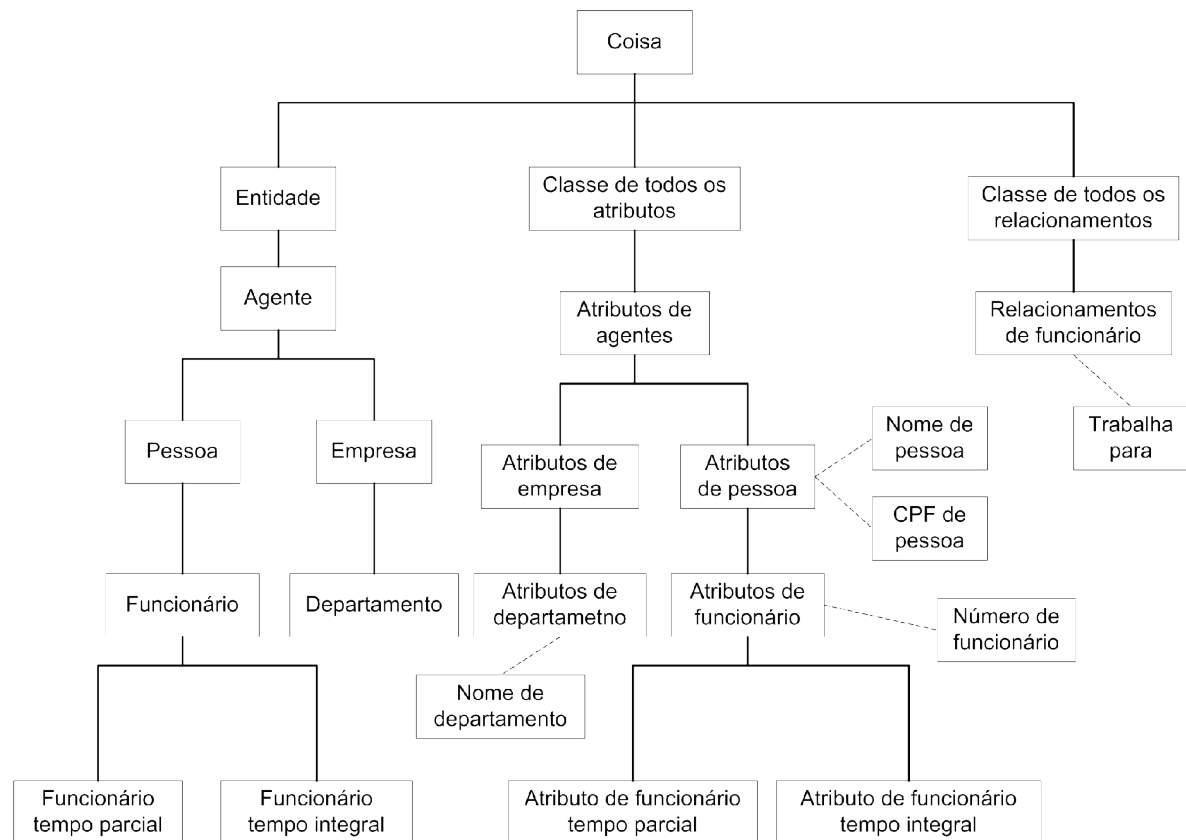


Figura 8 Exemplo de uma ontologia para uma empresa simples, mostrando classes e suas subclasses, relacionamentos e instâncias (indicadas pela linha tracejada)

Fonte: Huhns e Stephens (1999)

As palavras do vocabulário das palavras, usadas na linguagem de conteúdo, para representar objetos, conceitos e relacionamentos na área de interesse da mensagem. Uma ontologia pode ser representada em linguagem de lógica de predicados, para ser usada em conjunto com uma linguagem de conteúdo como KIF. Uma ontologia consiste das classes de objetos, suas subclasses, e de relacionamentos entre elas. Todos os agentes que compartilhem uma mesma ontologia possuem o mesmo entendimento do significado das palavras contidas nesta ontologia. Um exemplo de ontologia pode ser visto na figura . Esta figura representa uma ontologia simples para uma empresa. Ontologias necessitam representar as classes de

objetos e relacionamentos entre eles, mas não necessariamente todas as instâncias dos objetos. Na figura, as instâncias são representadas através de um relacionamento com linha tracejada com a classe da qual é instância (HUHNS; STEPHENS, 1999).

### 3.5.3.2 Padrão FIPA

FIPA (*Foundation for Intelligent Physical Agents* – <http://www.fipa.org>) é um grupo para a criação de especificações para a tecnologia de agentes. As especificações que produz padronizam vários aspectos da criação e operação de sistemas multiagentes. Entre elas estão (GARCIA, SICHMAN, 2003):

- arquitetura FIPA: descrição dos componentes da arquitetura e guia para a construção de agentes e plataformas;
- gerenciamento de agentes: como agentes interagem entre si;
- linguagem de comunicação de agentes: linguagem de alto nível para comunicação entre agentes;
- integração de agentes e software: serviços de suporte à integração de sistemas de agentes com outros sistemas;
- transporte de mensagens de agentes: especifica protocolos de comunicação a serem usados pelos agentes;
- serviços de ontologias: agentes trabalham com ontologias explicitadas de forma declarativa;
- interação homem-agente: especifica comunicações entre humanos e agentes;
- linguagens de conteúdo de agentes: especifica linguagens para transportar o

conteúdo das mensagens entre agentes.

### **3.6 Sistemas multiagentes existentes**

Nesta seção, serão apresentados dois *frameworks* para a criação de sistemas multiagentes existentes, o RETSINA e o JADE. Estes *frameworks* possibilitam a criação de sistemas multiagentes baseados em uma plataforma existente, que implementa os serviços básicos presentes em uma arquitetura multiagentes, como comunicação e mobilidade.

#### **3.6.1 RETSINA**

RETSINA (*Reusable Environment for Task-Structured Intelligent Networked Agents*) é uma arquitetura de sistemas multiagentes desenvolvida pelo Laboratório de Agentes de Software do Instituto de Robótica da Universidade Carnegie Mellon, nos Estados Unidos. Esta arquitetura consiste de quatro tipos básicos de agentes (RETSINA, 2001):

- agentes de interface: interagem com o usuário;
- agentes de tarefas: ajudam o usuário a realizar tarefas;
- agentes de informação: provêm acesso a fontes heterogêneas de informação;
- agentes de mediação: ajudam agentes que necessitam de serviços a encontrar agentes que disponibilizam serviços.

Cada agente na arquitetura RETSINA possui quatro módulos reutilizáveis para a

comunicação, planejamento, agendamento e monitoramento da execução das tarefas do agente. Em adição a estes módulos padrão, existem módulos que implementam as tarefas e visão do ambiente para o agente.

Os agentes na arquitetura RETSINA implementam um princípio de comunidade de agentes que se comunicam entre si. A estrutura do sistema multiagentes advém das várias interações entre os agentes, e não de uma estrutura rigidamente estabelecida. As funcionalidades previstas na arquitetura são(RETSA, 2001; DECKER; SYCARA, 1997):

- interoperação dos sistemas multiagentes;
- mapeamento de capacidade dos agentes;
- mapeamento de localização dos agentes;
- segurança;
- serviços de monitoramento de desempenho;
- serviços de gerenciamento do sistema;
- infraestrutura de linguagem de comunicação de agentes;
- infraestrutura de comunicação;
- ambiente operacional.

### 3.6.2 JADE

JADE (*Java Development Framework*) é um framework, implementado na linguagem Java, desenvolvido pelo laboratório TILAB, laboratório da Telecom Italia. É um framework distribuído como código aberto, e implementa as especificações FIPA (JADE, 2005).

O objetivo da criação do framework é simplificar a tarefa de desenvolver e executar agentes inteligentes. JADE inclui uma plataforma que executa os agente e uma biblioteca de componetes para o desenvolvimento dos agentes. A arquitetura JADE provê serviços de gerenciamento e comunicação de acordo com a especificação FIPA, uma interface de controle e observação dos agentes em execução, um ambiente de execução multiplataforma, bibliotecas para (CAIRE, 2002):

- criação de agentes;
- criação e envio de mensagens;
- definição de comportamentos;
- ativação dos agentes e de seus comportamentos;
- protocolos de interação;
- ontologias para a comunicação.

Agentes são criados através da extensão de classes Java já existentes, assim como mensagens e comportamentos. O ambiente de execução implementa serviços de (CAIRE, 2002):

- gerenciamento de nomes;
- gerenciamento de mensagens;
- serviços de “páginas amarelas” para comunicação entre agentes;
- mobilidade de agentes.

A plataforma JADE já possui um conjunto de extensões, com módulos existentes para integrá-lo com outras bibliotecas e frameworks em código aberto para a implementação de agentes inteligentes (JADE, 2005).

### **3.7 Técnicas de inteligência artificial**

Nesta seção, serão exploradas brevemente algumas técnicas de inteligência artificial, especialmente as relacionadas à análise de textos, que serão utilizadas na modelagem do sistema proposto.

#### **3.7.1 Programação por demonstração**

Programação por demonstração (PBD – *Programming By Demonstration*), ou programação por exemplo, consiste em instruir um sistema de software através de demonstrar, em um conjunto de dados de exemplo, como o sistema deve agir. O sistema então generaliza as ações requeridas para um conjunto mais amplo de dados (LIEBERMAN, 2005).

Através da programação por exemplo, é retirada da atividade de programar a necessidade de pensamento abstrato e de boa memória de curto prazo. O agente observa as ações do usuário e tenta repeti-las. Ao contrário de uma simples macro, que repete de modo idêntico as ações do usuário, um agente implementando PBD possui a capacidade de generalizar os passos da ação, tornando-o mais útil e flexível. Programar um software através de exemplos é também mais fácil para os usuários, já que o aprendizado nos seres humanos também é através da observação do comportamento de outros (LIEBERMAN, 2002).

Existem duas formas distintas de instruir um programa através de demonstração, instrução direta e inferência indutiva. Na instrução direta, o usuário indica ao sistema que está



executando um exemplo, e o sistema observará suas ações e então tentará generalizar um procedimento aplicável a outros exemplos semelhantes no futuro. Na inferência indutiva, o sistema observa continuamente o comportamento do usuário e decide autonomamente quais as ações são um exemplo e devem ser repetidas em oportunidades futuras (LIEBERMAN; MAULSBY, 1996).

### 3.7.2 Mineração de textos

Mineração de textos (*text mining* em inglês) é o nome dado às várias técnicas que visam à extração de informação de documentos não estruturados, escritos em linguagem natural. Existem duas abordagens principais quanto à mineração de textos, a análise semântica baseada na função das palavras no texto e a análise estatística que considera a frequência das palavras (EBECKEN, LOPES, COSTA, 2003).

Chama-se mineração de textos o processo de extração de padrões interessantes e não triviais, ou conhecimento, de documentos não estruturados. A mineração de textos é um campo multidisciplinar, envolvendo várias áreas da inteligência artificial no esforço de extrair informações de textos altamente não estruturados escritos em linguagem natural (TAN, 99).

A atividade de mineração de textos inclui como uma das etapas iniciais a preparação dos dados a serem analisados. Essa etapa de preparação inclui a redução de dimensionalidade, ou seja, reduzir o número de termos diferentes presentes no texto. Existem duas maneiras de se reduzir o número de termos, a remoção de *stopwords* e o *stemming*. Por *stopwords* entende-se palavras que não contribuem com significado ou conhecimento no texto, tais como artigos, preposições ou pronomes. *Stemming* é o processo que diminui a quantidade de termos no

texto através do agrupamento de palavras semelhantes. Este agrupamento é obtido através da redução das palavras isoladas à sua palavra raiz, pela remoção do seu sufixo (EBECKEN, LOPES, COSTA, 2003).

Dentre os métodos de *stemming* existentes, destacam-se o método do Stemmer S, o método de Porter e o método de Lovins. Estes métodos foram desenvolvidos para a língua inglesa, mas existem adaptações dos mesmos para várias outras línguas, inclusive o português (EBECKEN, LOPES, COSTA, 2003).

O método Stemmer S consiste da remoção dos sufixos que determinam o plural (ies, es, s). É considerado um sistema conservador e previsível, apesar de não remover muitos sufixos. O método de Porter identifica as diferentes inflexões de uma mesma palavra e as substitui por um radical comum. Este método remove 60 sufixos diferentes. O método de Lovins foi desenvolvido a partir de um conjunto de exemplos de palavras, e consegue remover aproximadamente 250 sufixos diferentes (EBECKEN, LOPES, COSTA, 2003).

A redução de dimensionalidade também pode ser obtida através da utilização de um dicionário ou thesaurus. Um dicionário consiste em um vocabulário controlado, através do qual os termos no texto são indexados. Esta abordagem também permite o agrupamento de sinônimos que não possuem a mesma raiz, assim como, em processos mais sofisticados, o relacionamento entre termos diferentes (EBECKEN, LOPES, COSTA, 2003).

O processo de mineração de textos pode consistir de várias tarefas que buscam extrair conhecimento ou informação de documentos não estruturados. Cada tarefa extrai um tipo de informação diferente do texto. Entre as tarefas utilizadas, tem-se (EBECKEN, LOPES, COSTA, 2003):

- clustering: também chamado agrupamento, busca juntar documentos semelhantes;
- categorização: busca classificar documentos em categorias pré-definidas;
- indexação: gera um índice de forma que se possa buscar palavras no texto sem que

seja necessário varrer todo o documento;

- extração de características: extrai informações específicas do texto, tais como nomes de pessoas e endereços;
- sumarização: concentra-se em criar um resumo do texto, tentando extrair a principal idéia do mesmo.

A seguir serão analisadas as tarefas mais significativas ao sistema multiagentes proposto, com uma breve discussão de técnicas existentes na literatura para levá-las a cabo.

### **3.7.3 Análise estatística de texto**

A análise estatística de textos parte do princípio que a importância de um termo para a análise de um documento pode ser dada pela frequência com que este termo aparece no texto. Esta técnica é em grande parte independente do idioma do texto a ser analisado. Os passos principais desta técnica incluem (EBECKEN, LOPES, COSTA, 2003):

- codificação dos dados: corresponde à preparação do documento para a análise, com a redução da dimensionalidade do mesmo e outras técnicas de preparação para melhorar o desempenho dos passos seguintes;
- estimativa dos dados: consiste da aplicação de um método de estimativa ou aprendizagem estatística para obter um modelo dos dados;
- modelos de representação de documentos: é a forma como o documento é representado dentro do sistema. A mais comum é chamada de modelo *bag o words*, que ignora a ordem das palavras no texto e foca somente em sua frequência.

Modelos estatísticos de recuperação de informação em textos muito utilizados atualmente implementam alguma variante de um Modelo de Espaço Vetorial (*Vector Space Modeling* – VSM). Um modelo VSM representa os documentos como vetores, com cada posição representando um termo e cada valor representando a frequência na qual o termo aparece. O modelo mais simples emprega um vetor binário, e só permite consultas sobre a pertinência ou não do termo no documento. Em modelos mais sofisticados, como o Modelo Ponderado por Termos (*Term-Weighted Model*), o vetor é de valores reais, e representa não só a existência do termo no documento, mas a sua frequência, a posição relativa do termo, entre outras informações. Um modelo ponderado busca melhorar a representação da importância relativa de um termo em relação aos outros no documento (KOBAYASHI, AONO, 2004).

Uma análise de vários algoritmos estatísticos de categorização, usando o corpus de documentos da Reuters, mostrou que algoritmos que aprendem têm uma performance vastamente superior aos algoritmos que não apresentam aprendizado (YANG, 1999).

Existem diversas abordagens para a análise estatística, que serão expostas brevemente a seguir. Entre elas encontram-se: support vector machines (SVM), redes bayesianas e *hidden Markov models* (HMM).

Support Vector Machines (SVM) são sistemas usados para classificar elementos em categorias, possuindo aprendizado através da apresentação de um conjunto de treinamento que determina os parâmetros do sistema. SVM criam um espaço multidimensional no qual os elementos do conjunto de treinamento são plotados. O sistema passa então a determinar hiperplanos que dividem os diversos grupos de elementos de treinamento e, para maximizar a precisão do classificador, os hiperplanos se posicionam de modo a maximizar a distância deles com os pontos de treinamentos de cada grupo (CHAKRABARTI, 2003).

Redes Bayesianas também são usadas para classificar documentos com base em estatísticas de termos. Depois de criada a estrutura da rede e com um conjunto de treinamento,

é simples extrair as relações probabilísticas. O problema consiste na geração da estrutura, especialmente quando o número de nós é grande, pois a complexidade cresce muito rapidamente. A técnica só é usada para a classificação de documentos entre duas categorias, pois a complexidade computacional e o tamanho das tabelas de probabilidades cresceriam muito (CHAKRABARTI, 2003).

Rede Bayesiana é um modelo baseado em grafos que representa relações probabilísticas entre atributos. O grafo é um Grafo Direcionado Acíclico (Directed Acyclic Graph – DAG), no qual os nós representam os atributos e os arcos representam os modelos probabilísticos que conectam os atributos. Os modelos de probabilidade possibilitam que o sistema lide com incertezas, e permite que o sistema infira os demais atributos baseado no valor de um subconjunto de atributos (RAYNOR, 1999; LUSTOSA, 2004).

A topologia de uma rede bayesiana pode ser considerada como uma base de conhecimento que contém uma variedade de informações diferente, pois ela representa as relações entre os atributos do universo em estudo, e não detalhes sobre cada atributo. Várias informações sobre os detalhes dos atributos estão implícitas na probabilidade dos relacionamentos entre eles (RUSSEL, NORVIG, 1995).

Hidden Markov Model é um modelo estatístico, em que o sistema sendo modelado possui parâmetros escondidos. Um modelo de Markov é um modelo consistindo de uma sequência de estados, com a probabilidade de uma transição de um estado para outro dependendo somente do estado atual, e não dos estados anteriores (HIDDEN, 2005).

Um modelo Hidden Markov, assim como um modelo de Markov, descreve processos que seguem uma sequência de estados. Num modelo de Markov tradicional, a saída é uma sequência estados. Num HMM, cada estado tem uma distribuição probabilística de saídas, e a mesma saída pode existir em mais de um estado. Um observador não sabe a sequência de estados, pois quando o modelo produz uma saída, não se pode determinar com certeza qual

estado a gerou (RUSSEL, NORVIG, 1995).

### **3.7.4 Análise de linguagem natural**

A análise de linguagem natural parte de conhecimentos lingüísticos para a obtenção de informações do texto. Isso inclui conhecimentos morfológicos, sintáticos, semânticos, pragmáticos, de discurso e sobre o domínio a ser estudado (EBECKEN, LOPES, COSTA, 2003).

Conhecimentos morfológicos são informações sobre a estrutura, forma e inflexões das palavras. Conhecimentos sintáticos abordam como as palavras são combinadas para produzir sentenças. Conhecimentos semânticos são sobre os significados das palavras e como palavras podem ser combinadas para criar novos significados. Conhecimentos pragmáticos abordam o uso da língua em diferentes contextos e como o contexto afeta a interpretação. Conhecimentos do discurso tratam de como sentenças próximas afetam o significado de uma sentença. Conhecimentos do domínio são informações sobre o domínio a que a comunicação se refere (EBECKEN, LOPES, COSTA, 2003).

Pode-se considerar dois tipos de linguagem, a linguagem formal e a linguagem natural. Por linguagem formal entende-se uma linguagem como de programação, na qual as palavras possuem significados determinados e não ambíguos, a linguagem possui um sintaxe fixa e um vocabulário definido. Linguagens naturais são mais complexas, pois uma mesma palavra pode significar coisas diferentes, e mesmo uma sentença pode significar coisas diferente em diferentes contextos. Essa grande complexidade das linguagens naturais torna difícil a interpretação de um texto por um sistema computacional (RUSSEL, NORVIG, 1995).

Uma forma de simplificar este problema é analisar somente um subconjunto da linguagem, com regras claras que a aproximem de uma linguagem formal. Essa linguagem teria um vocabulário restrito, com significado não ambíguos, e uma gramática também definida (RUSSEL, NORVIG, 1995).

Partindo de uma gramática, pode-se analisar (*parse*) a frase e identificar os termos constituintes como nomes, substantivos, verbos, artigos, pronomes, etc. A partir dessa identificação, precisa-se retirar o significado da sentença. Pode-se considerar as informações semânticas como expressões de lógica de predicados de primeira ordem. A próxima tarefa é partir de uma sentença cujos termos estejam identificados segundo uma gramática e traduzí-los para lógica de predicados, que possa ser processada pelo agente (RUSSEL, NORVIG, 1995).

Os sistemas existentes de análise de linguagem natural funcionam bem, porém nenhum deles consegue exibir uma boa performance em um domínio aberto. Fazer a interpretação correta de uma sentença requer conhecimentos de várias fontes, incluindo sintática, lexical e semântica. A análise das sentenças adjacentes fornece subsídios à escolha da interpretação correta do significado de uma sentença (RUSSEL, NORVIG, 1995).

Historicamente, o trabalho de Processamento de Linguagem Natural (Natural Language Processing – NLP) tem se focado no uso de *parsers* que utilizam um vocabulário delimitado e regras gramaticais para montar a estrutura das sentenças (WATSON, 2004).

Existem várias técnicas para a criação de sistemas NLP (WATSON, 2004):

- máquina de estados para determinar se uma sequência de palavras é uma sentença sintaticamente válida;
- analisadores gramaticais de dependência conceitual, que focalizam mais a semântica do que a sintaxe; esta técnica não é mais tão popular;
- técnicas estatísticas de NLP.

Uma máquina de estados usada para analisar uma sentença é chamada de *parser* ATN (augmented transition networks). Os estados correspondem às diferentes funções sintáticas das palavras, com uma sequência de estados determinando uma estrutura de sentença. Um parser ATN consiste de um conjunto de estrutura de sentenças, que são analisados e comparados à sentença sendo processada até que a estrutura correta seja encontrada. Existem outras técnicas para determinar a estrutura de uma sentença, como Hidden Markov Models e redes Bayesianas (WATSON, 2004).

Processamento estatístico de linguagem natural consiste em usar um método estatístico para determinar a função sintática das palavras e a estrutura das sentenças. Um método utiliza Hidden Markov Models (HMM) para analisar um conjunto de sentenças (texto). Um modelo HMM transforma uma entrada, consistindo de um conjunto de palavras, em uma saída consistindo de um conjunto de atribuições morfológicas para estas palavras. Este modelo é inicialmente treinado com textos no qual a estrutura das sentenças esteja previamente estabelecida. Depois de treinado, o algoritmo calcula a função sintática de uma palavra baseado na probabilidade de uma palavra possuir esta função e na probabilidade de uma determinada função ocorrer após outra (WATSON, 2004).

### **3.7.5 Clustering e Categorização**

Clustering e categorização são técnicas de análise de texto que separam um conjunto de documentos de acordo com similaridades entre eles.

Clustering é uma técnica que agrupa documentos, baseando-se nas similaridades entre eles. Clustering não possui um conjunto de categorias pré-existente, através da análise



das similaridades entre os documentos o sistema determina as categorias, ou clusters, existentes (RAYNOR, 1999).

A utilidade de clustering para a mineração de textos advém da hipótese de que se um usuário está interessado em um documento  $d$ , também estará interessado nos outros documentos pertencentes ao cluster de  $d$  (CHAKRABARTI, 2003).

Categorização é uma técnica que agrupa documentos em categorias, que são definidas previamente. O sistema analisa o documento e determina a quais categorias o documento pertence (RAYNOR, 1999).

Uma ferramenta de categorização é inicialmente treinada com um conjunto de documentos que já estão associados a uma categoria. Depois de treinada, a ferramenta é capaz de associar uma categoria a um documento desconhecido, baseando-se nas similaridades entre este documento e as características das categorias (CHAKRABARTI, 2003).

### **3.7.6 Redes neurais**

Uma rede neural procura imitar o processo que os seres vivos possuem através da criação de uma rede de unidades de processamento simples, chamadas neurônios, interligadas (RUSSEL, NORVIG, 1995).

Uma rede neural artificial é um conjunto de unidades de processamento simples que trocam sinais entre si através de muitas conexões ponderadas. Uma rede neural possui (KRÖSE, VAN DER SMAGT, 1996):

- um conjunto de unidades de processamento – neurônios;
- uma ativação para cada neurônio, que representa a saída da rede;

- conexões entre os neurônios, com um respectivo peso, que representam a influência de um neurônio sobre outro;
- uma regra de propagação, que determina o sinal de entrada de um neurônio baseado nos sinais individuais que ele recebe;
- uma função de ativação, que determina a ativação de um neurônio baseada no sinal de entrada e na ativação anterior do neurônio;
- um sinal externo (bias) para cada neurônio;
- um método de aprendizado;
- um ambiente no qual o sistema opera, que fornece as entradas e, possivelmente, feedback das saídas.

Uma rede neural é um sistema altamente paralelo de processamento, com capacidade de aprendizado e de generalização. As principais características de uma rede neural são (HAYKIN, 1994 ):

- não linearidade: um neurônio é um sistema não linear, e conseqüentemente a rede formada pelos neurônios é não linear;
- mapeamento entrada-saída: através do processo de treinamento de uma rede, ela pode realizar um mapeamento entre um conjunto de entradas e um conjunto de saídas;
- adaptabilidade: redes neurais pode mudar os pesos das sinapses para adaptar-se a mudanças nas condições operacionais;
- resposta com evidência: em um caso de classificação de padrões, uma rede neural pode ser construída de modo a não só classificar a entrada, mas também informar o nível de confiança na decisão;
- informação contextual: o conhecimento é representado de forma distribuída em toda a estrutura da rede, nos pesos das sinapses e nas ativações dos neurônios;

- tolerante a falhas: como a informação é armazenada de modo distribuído ao longo de toda a rede, a falha de um neurônio afeta a qualidade do processamento, mas não o impede nem torna a rede incapaz de operar.

## 4 Descrição do modelo proposto

Este capítulo descreve o modelo proposto da arquitetura de agentes, baseando-se nas necessidades e requisitos levantados no estudo do processo de inteligência competitiva e chegando a uma descrição da proposta usando agentes inteligentes. A proposta é conceitual, e descreve principalmente quais as tarefas que os diferentes tipos de agentes devem realizar, e as técnicas necessárias para implementá-las.

Na primeira parte do capítulo, a arquitetura básica proposta é exposta, seguida da descrição das interações básicas previstas com o usuário. Após, são discutidos os tipos básicos de agentes contidos na proposta, a estrutura dos mesmos e algumas propostas de mensagens usadas entre eles.

### 4.1 *Arquitetura básica*

Como pode ser visto em Silva (2000), a principal função do modelo proposto é o monitoramento de fontes conhecidas de informação na internet e a busca de novas fontes, automaticamente informando o usuário de novas páginas ou sites que se relacionem às suas necessidades de informação. A partir destes requisitos básicos, partiu-se para a elaboração da proposta do sistema de agentes.

Estruturalmente, a proposta prevê a existência de cinco tipos básicos de agentes, com subtipos mais especializados, conforme pode ser visto na figura 9. Nesta figura, um diagrama

da UML (*Unified Modeling Language* – Linguagem de Modelagem Unificada), cada tipo de agente é representado por uma classe, e os subtipos por uma classe derivada (subclasse).

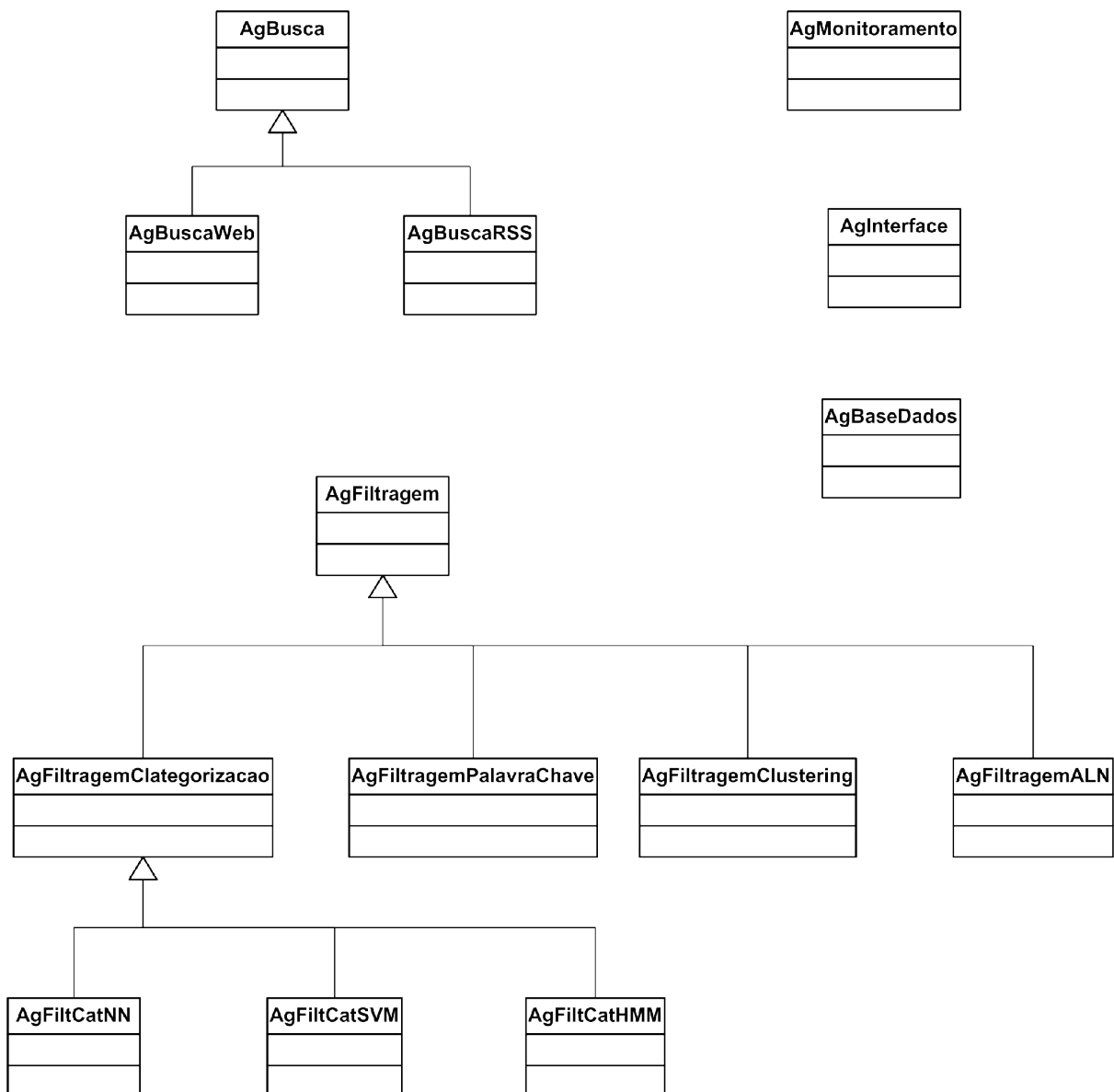


Figura 9 Visão geral dos tipos de agentes propostos

Um dos princípios na criação da proposta foi a decomposição de tarefas entre os agentes, para diminuir a complexidade e necessidade de recursos em cada agente, conforme

Huhns e Stephens (1999). A decomposição das tarefas foi então feita com agentes de interface, agentes de busca, agentes de monitoramento, agentes de filtragem e agentes de base de dados.

O agente de interface tem como tarefas interagir com o usuário e controlar a ativação dos demais agentes. Os agentes de busca têm como tarefa buscar novas páginas que satisfaçam o critério do usuário. Os agentes de monitoramento têm como tarefa a verificação periódica dos sites conhecidos em busca de atualizações. Os agentes de filtragem têm como tarefas a análise das páginas recuperadas pelos agentes de busca e monitoramento, utilizando técnicas de mineração de textos, buscando determinar a importância da página recuperada em relação às especificações do usuário. O agente de base de dados tem como tarefa prover serviços de banco de dados para os demais agentes, tanto para armazenar os textos recuperados quanto informações necessárias para as tarefas dos agentes, proporcionando uma camada de abstração que permite que os agentes não necessitem capacidades e acesso direto ao banco de dados, seguindo a prática de encapsulamento das técnicas de programação orientadas a objetos, segundo Eckel (2002) e Meyer (2000).

## ***4.2 Interação com o usuário***

Do ponto de vista do usuário, a interação com o sistema proposto pode ser dividida em três atividades ou tarefas. Estas tarefas podem ser vistas na figura 10, que as representa em um diagrama de casos de uso (use cases) na linguagem UML. As tarefas são entrada de dados, envio do relatório (pelo sistema) e feedback (retroalimentação) dos agentes.

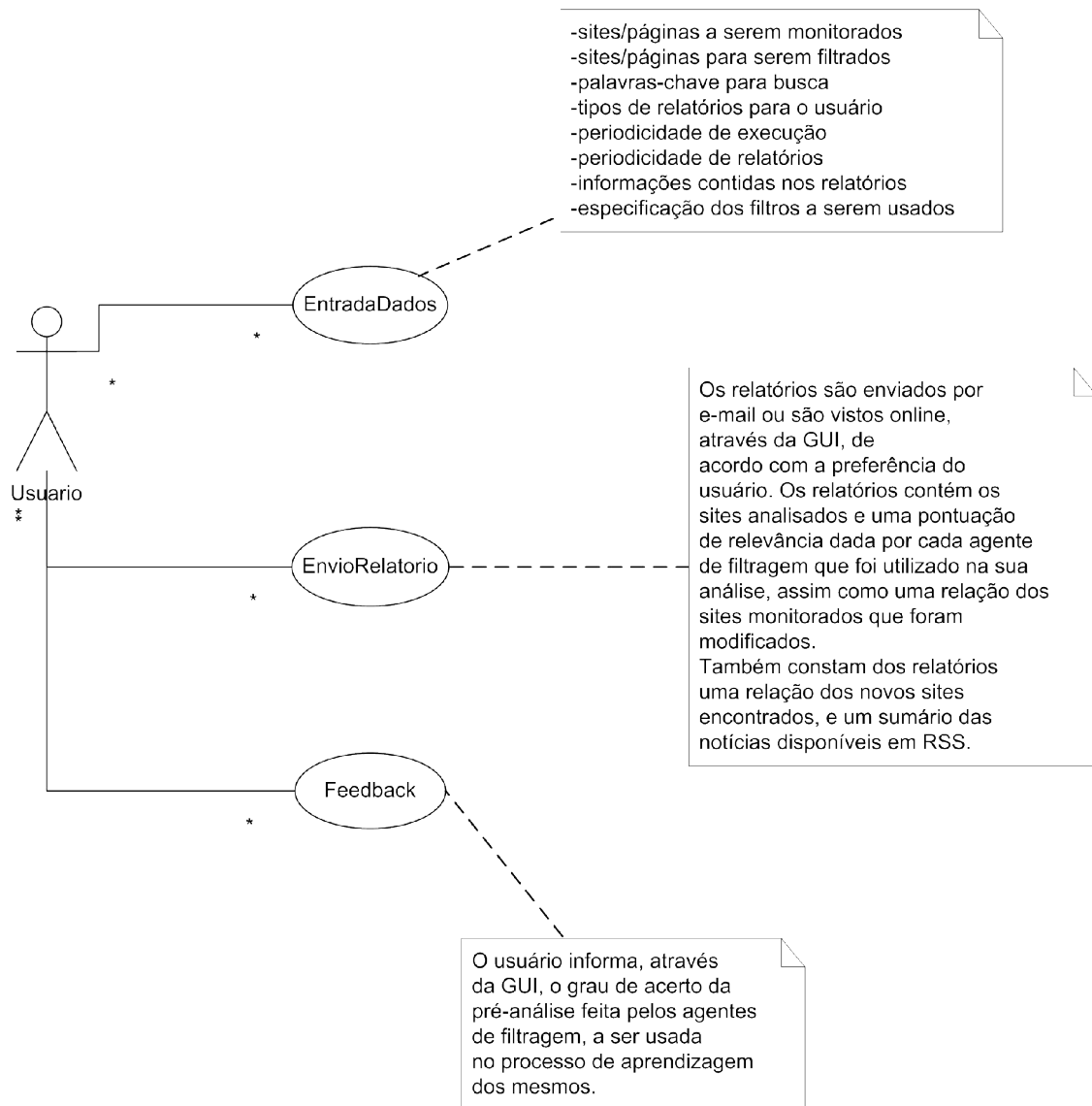


Figura 10 Casos de uso representando a interação com o usuário

O caso de uso de EntradaDados agrupa todas as informações que o usuário fornece aos agentes:

- sites e páginas a serem monitorados;
- sites e páginas a serem filtrados;
- palavras-chave para a busca;
- tipos de relatórios a serem fornecidos ao usuário;

- periodicidade de execução dos agentes;
- periodicidade dos relatórios;
- informações contidas nos relatórios;
- especificação dos filtros a serem usados.

O caso de uso *EnvioRelatorio* engloba a resposta que o sistema proposto dá ao usuário, os alertas e relatórios que este recebe.

O caso de uso *Feedback* engloba a avaliação, pelo usuário, do desempenho do sistema, em especial da qualidade dos resultados da filtragem. Este retorno é usado para implementar a aprendizagem nos agentes. Agentes que aprendem possuem um desempenho superior em comparação com os que não aprendem, conforme Yang (1999).

### **4.3 Tipos de agentes**

A figura 9 mostra um diagrama dos diferentes tipos de agentes. Neste diagrama UML, os agentes são representados como classes. Os agentes de busca e agentes de filtragem, por existirem em mais de um tipo, possuem classes derivadas (subclasses).

A idéia inicial do agente de busca, contida em Silva (2000), é a de um robô que navegue pela web buscando novos sites de interesse. Esta funcionalidade é muito semelhante à provida pelos sistemas de busca comerciais que existem atualmente, tais como Google, Altavista, Yahoo, entre outros. Estes sistemas de busca têm fornecido APIs (*Application Program Interface* – Interface de Programa Aplicativo) que permitem o acesso direto aos motores de busca destes sistemas. Estas APIs constituem-se de bibliotecas de programação



que permitem que um software pesquise diretamente no sistema de buscas, e uma interface no sistema de buscas que receba, interprete e envie respostas aos programas criados pelas APIs. No momento do desenvolvimento desta proposta, tanto Google quanto Yahoo fornecem essas APIs. (GOOGLE WEB APIs., 2005; YAHOO! Search Developer Network., 2005). Com base neste fato, foi decidido propor agentes de busca que utilizassem os serviços já existentes de busca na Web, já que estes serviços possuem um grande repositório de páginas indexadas e sofisticados algoritmos de busca, que não poderiam ser duplicados com facilidade em um agente de busca. Desse modo, o resultado da tarefa do agente de busca é uma lista de sites, fornecida pelo sistema de busca acessado.

Uma nova tecnologia para distribuição de conteúdo surgiu em 1999, o RSS (*Really Simple Syndication* – Distribuição Realmente Simples). É uma tecnologia que consiste de um documento XML, em um servidor WEB, onde é definido um canal de comunicação com diversos itens de informação, que são lidos pelo usuário usando um programa leitor de RSS. O canal de informação informa ao software de leitura o site de onde o RSS se origina, servindo como um container para os itens de informação. Cada item de informação consiste de um título, um link e uma descrição. O uso de RSS é mais indicado para sites que são atualizados com frequência, pois o leitor de RSS automaticamente acessa o site e baixa os itens de informação. Isso libera o usuário da necessidade de acessar o site regularmente somente para verificar se existem atualizações (RSS TUTORIAL, 2005). O fato de a tecnologia RSS estar sendo adotada por uma grande quantidade de sites, especialmente sites que publicam conteúdo periodicamente, a torna interessante para o sistema proposto, pois o monitoramento de informações em tais sites é um requisito primário. O resultado de uma recuperação de documentos RSS é uma lista de links, com títulos e descrições. Este formato é o mesmo retornado pelo agente de busca quando acessa um sistema de busca como o Google. Baseando-se nesta semelhança de resultado e de propósito, decidiu-se modelar os agentes de

busca em dois tipos, um capaz de acessar os sistemas de busca como o Google ou o Yahoo, e outro capaz de acessar documentos RSS disponibilizados por sites a serem monitorados.

Desse modo, o objetivo primário dos agentes de busca é retornar uma lista de endereços que correspondam a uma palavra chave (ou conjunto de palavras chave) fornecida pelo usuário, através do acesso a um sistema de busca já existente na internet, como o Yahoo ou o Google. Ou retornar uma lista de atualizações, na forma de documentos RSS, através do acesso aos sites, cadastrados pelo usuário, que forneçam este serviço.

Utilizando a tipologia de agentes mostrada na figura 4, pode-se classificar os agentes de busca como: segundo a cognitividade, agentes reativos, pois executam a sua tarefa e simplesmente respondem à sua programação e ao ambiente (sites) no qual atuam; segundo a mobilidade, agentes imóveis, pois apesar de atuarem na Internet, seu local de execução permanece sempre o da máquina de origem; segundo a comunicabilidade, são agentes comunicativos, pois trocam mensagens com os demais agentes do sistema para realizar suas tarefas; segundo a aprendizagem, são agentes sem aprendizado, pois suas tarefas são realizadas sempre da mesma forma.

O agente de interface, no modelo proposto, é o responsável por apresentar ao usuário uma interface gráfica com o sistema, através da qual o usuário entra com as diversas informações necessárias para o funcionamento dos agentes e recebe as informações resultantes das atividades dos mesmos. Outra função do agente de interface, sugerida inicialmente por Silva (2000), é a de controlador das atividades e prioridades dos demais agentes. Ele também possui o papel de mediador entre os demais agentes e o usuário, intermediando a troca de informações entre eles. Em uma visão organizacional do sistema, como a vista em Garcia e Sichman (2003), o agente de interface faria o papel do chefe, determinando qual agente seria ativado e quando isso ocorreria. Através da criação de um agente de interface para cada usuário, ou mesmo para cada atividade de inteligência independente de um usuário, pode-se

atender ao requisito de manter um perfil de necessidades de informação, no caso representado principalmente por palavras chave e categorias de classificação de documentos, para cada usuário, como descrito em Silva (2000). O agente de interface também deve possuir a capacidade de montar um relatório com o resultado das atividades dos outros agentes. Este relatório pode ser acessado pela interface gráfica do agente, ou ser recebido pelo usuário via e-mail, conforme requisito levantado por Silva (2000). Na implementação desta funcionalidade, as técnicas de programação por demonstração podem ser aplicadas para permitir ao usuário especificar o formato e conteúdo dos relatórios desejados (LIEBERMAN, 2005).

Ao classificar-se os agentes de interface segundo a tipologia apresentada na figura , tem-se: segundo a cognitividade, são agentes cognitivos, pois utilizam-se de uma representação das necessidades de informação do usuário para determinar as ativações dos demais agentes e preencher os relatórios; segundo a mobilidade, são agentes imóveis; segundo a comunicabilidade são agentes comunicativos, pois se comunicam com os demais agentes dentro do sistema para realizar suas tarefas; segundo a aprendizagem, são agentes com aprendizado, pois modificam seu comportamento para melhor realizar suas atividades, o aprendizado se dando através da entrada direta de instruções pelo usuário.

Os agentes de monitoramento, nesta proposta, possuem a tarefa de acessar sites e páginas isoladas a intervalos regulares, para determinar se o conteúdo destas páginas ou sites foi atualizado. Os sites a serem monitorados são alimentados pelo usuário, assim como a frequência de monitoramento de cada site. O resultado das atividades do agente é uma lista de sites atualizados que precisam ser analisados.

A classificação dos agentes de monitoramento segundo a tipologia da figura 4 é: segundo a cognitividade, são agentes reativos, pois simplesmente realizam suas atividades e respondem à configuração dos sites que monitoram; quanto à mobilidade, são agentes imóveis; quanto à comunicabilidade são agentes comunicativos; quanto à aprendizagem, são

agentes sem aprendizado, pois realizam o monitoramento sem modificar seu comportamento e seu desempenho no decorrer do tempo.

O agente de base de dados está presente na proposta para encapsular as atividades de acesso a base de dados, de forma a isolar dos demais agentes as funções de comunicação com o sistema de base de dados. Isso segue a noção de orientação a objetos de encapsular as capacidades dos módulos de um sistema, para simplificar o desenvolvimento e manutenção do software, como exposto por Eckel (2002) e Meyer (2000).

Quanto à tipologia de agentes representada na figura 4, a classificação dos agentes de base de dados é: quanto à cognitividade, são agentes reativos, pois suas tarefas se resumem a acessar o sistema de banco de dados, de forma automática e respondendo às requisições dos demais agentes; segundo a mobilidade, são agentes imóveis; segundo a comunicabilidade, são agentes comunicativos, por participarem de um sistema multiagentes; segundo a aprendizagem, são agentes sem aprendizado, pois seus comportamentos não se modificam no decorrer do tempo.

A proposta para os agentes de filtragem é ampla, prevendo a existência de vários tipos de agentes de filtragem, a serem desenvolvidos e implementados ao longo do tempo. Isso permite que novos métodos de análise de texto sejam incorporados no sistema a medida que este for evoluindo, e também permite que múltiplos métodos sejam usados em um mesmo texto, buscando fazer com que os pontos fracos dos métodos se cancelem e produzam um melhor resultado. Esta proposta veio da realização de que os algoritmos e métodos de análise de texto em uso atualmente possuem um desempenho difícil de quantificar e de prever, como pode ser visto em Yang (1999). Outro motivo para esta abordagem modular e extensível dos agentes de filtragem se dá pelo fato de, como o modelo proposto deve tratar de páginas web, o domínio do problema é bastante aberto, e o design modular procura prover adaptabilidade ao modelo. Os agentes de filtragem a serem utilizados em cada site devem ser especificados pelo

usuário, que recebe, no relatório dos resultados, o resultado de cada tipo de agente para um determinado site, o que permite que, através da experiência, o usuário possa determinar qual o agente de filtragem mais adequado para cada tipo de site a ser analisado.

A classificação dos agentes de filtragem, segundo a tipologia representada na figura , é: segundo a cognitividade, são agentes cognitivos, pois utilizam conhecimentos e técnicas de inteligência artificial para filtrar os sites; segundo a mobilidade, são agentes imóveis; segundo a comunicabilidade, são agentes comunicativos; segundo a aprendizagem, são agentes com aprendizado, pois adaptam seus parâmetros de filtragem de acordo com o feedback do usuário.

#### 4.4 Estrutura dos agentes

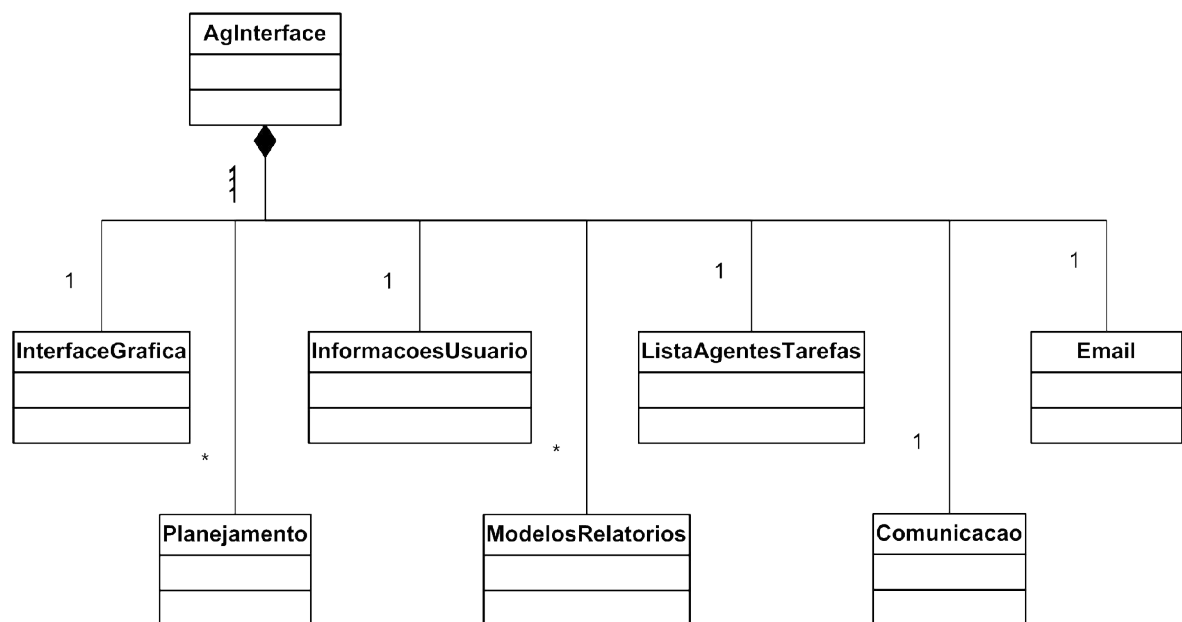


Figura 11 Proposta de estrutura do agente de interface

Os agentes propostos seguem a idéia de programação orientada a objetos, e decidiu-

se por uma proposta de arquitetura de agentes modular, na qual cada módulo corresponde a uma classe. A estrutura modular segue a linha de arquitetura de agentes defendida por Garcia e Sichman (2003), vista na seção 3.4 e figura 6. A seguir serão apresentadas as propostas para a estrutura de cada agente.

A estrutura básica proposta para os agentes de interface pode ser vista na figura 11.

Os módulos são:

- interface gráfica: chamado InterfaceGrafica na figura 11, este módulo provê as funcionalidades da interface gráfica com o usuário, tanto para receber informações do usuário quanto para apresentar os resultados. Também é responsável pela coordenação das atividades dos demais agentes e armazenamento das preferências do usuário;
- perfil de necessidades de informação: chamado InformacoesUsuario na figura 11, este módulo mantém informações que personalizam o comportamento do agente para cada usuário ou mesmo cada foco de inteligência competitiva de um usuário. Este módulo armazena informações sobre os sites que devem ser monitorados, palavras-chave, quais os agentes de filtragem a serem usados em cada site a ser monitorado, e outras informações que parametrizem a atividade de coleta. Também armazena informações sobre os relatórios a serem apresentados, com que modelo de relatório será utilizado e com que frequência;
- módulo de planejamento e tomada de decisão: chamado Planejamento na figura 11, este módulo implementa um algoritmo de inteligência artificial, a ser determinado, responsável por tomar as instruções do usuário armazenadas no módulo InformacoesUsuario e gerar um conjunto de instruções para os demais agentes, incluindo ordem de ativação, quantidade de agentes ativos em cada momento, assim como passagem de parâmetros de busca, monitoramento e

filtragem para os agentes apropriados;

- lista de tarefas a serem cumpridas: chamado ListaAgentesTarefas na figura 11, este módulo armazena o resultado do planejamento realizado no módulo Planejamento, sob a forma de uma lista de atividades a serem realizadas, incluindo entre estas as ativações e desativações dos demais agentes, a construção dos relatórios e o envio dos mesmos;
- modelos de relatórios: chamado ModeloRelatorios na figura 11, este módulo armazena os modelos de relatórios que podem ser gerados pelo agente;
- geração e envio de e-mail: chamado Email na figura 11, este módulo implementa as funções de envio de relatórios por e-mail;
- comunicação com outros agentes: chamado Comunicacao na figura 11, este módulo implementa as atividades de comunicação com os demais agentes do sistema, incluindo as linguagens utilizadas para a comunicação, os tipos de mensagens enviadas e recebidas e os protocolos de comunicação.

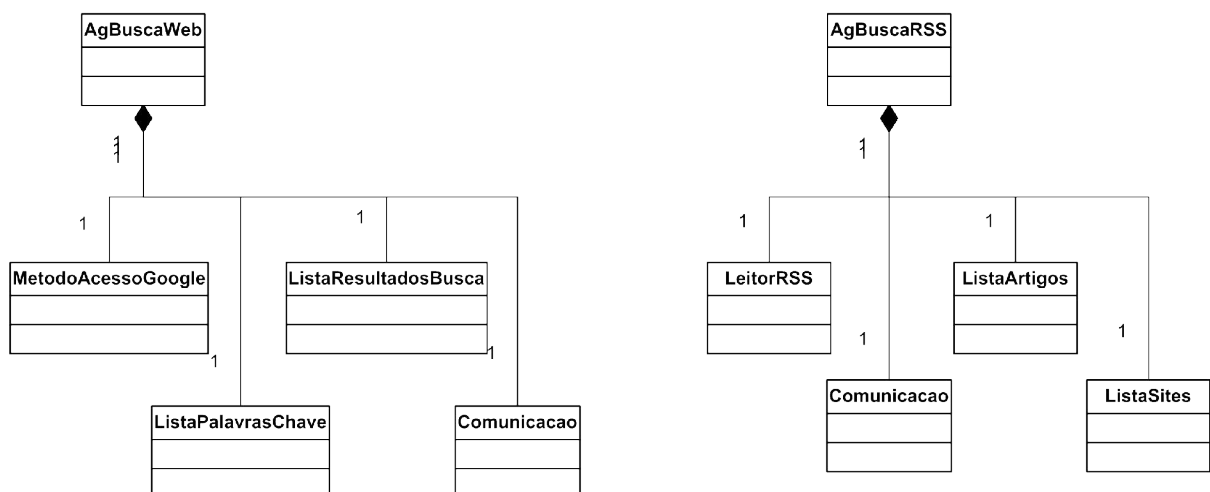


Figura 12 Proposta de estrutura para os agentes de busca

A estrutura dos agentes de busca pode ser vista na figura 12. Os agentes de busca na Web, chamados na figura de AgBuscaWeb, possui os seguintes módulos:

- acesso ao sistema de busca: chamado MetodoAcessoGoogle na figura 12, este

módulo implementa o acesso ao sistema de busca. No caso, propõem-se o uso do sistema Google, mas pelo fato de a proposta ser por um agente modular, pode-se utilizar outro sistema, como o Yahoo, bastando mudar o módulo de acesso;

- lista de palavras chave a serem buscadas: chamado ListaPalavrasChave na figura 12, este módulo armazena uma lista das palavras chave a serem usadas na busca;
- lista de resultado das buscas: chamado ListaResultadosBusca, este módulo armazena os resultados obtidos das consultas ao sistema de busca;
- comunicação: este módulo implementa a comunicação com os demais agentes.

Os agentes de busca de RSS, chamados, na figura 12, de AgBuscaRSS, apresentam os módulos:

- leitor de RSS: chamado LeitorRSS na figura 12, este módulo implementa os métodos de acesso a sites que disponibilizem RSS;
- lista de artigos monitorados: chamado ListaArtigos na figura 12, este módulo armazena os artigos que são recebidos dos sites, via RSS;
- lista de sites a serem acessados: chamado ListaSites na figura 12, este módulo armazena a lista dos sites a serem monitorados via RSS;
- comunicação: este módulo implementa a comunicação com os demais agentes.

A estrutura proposta para os agentes de monitoramento está representada na figura 13, na qual eles são chamados de AgMonitoramento. Os componentes dos agentes de monitoramento são:



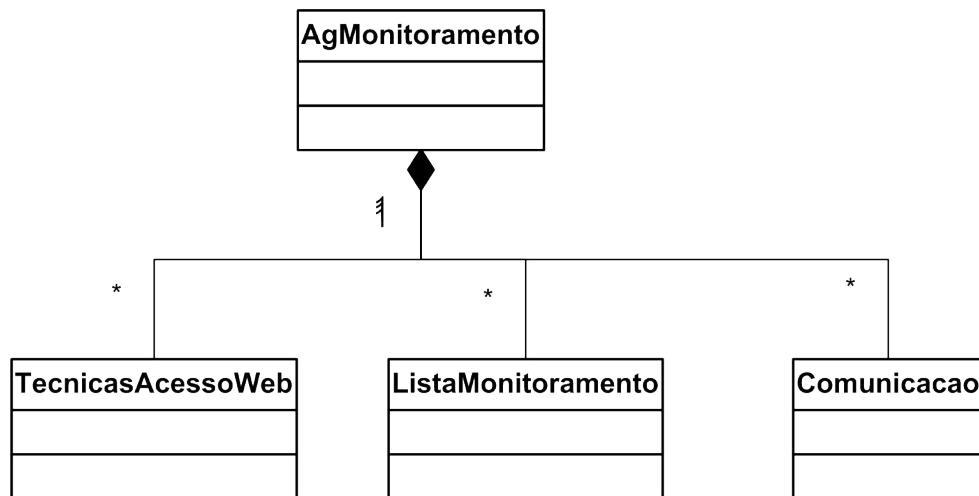


Figura 13 Proposta de estrutura para o agente de monitoramento

- acesso a páginas Web: chamado na figura 13 de *TecnicasAcessoWeb*, este módulo é responsável por implementar os métodos de conexão, parsing (extração de palavras) da página e comparação com uma imagem da página feita em acesso anterior. Esta imagem é o resultado da aplicação de uma função hash no texto da página. Esta função hash torna a descoberta de modificações a simples aplicação de uma função, pois qualquer modificação no texto alteraria o resultado; (CORMEN et al., 2001; LAFORE, 1998)
- lista de sites a serem monitorados: chamado de *ListaMonitoramento* na figura 13, este módulo armazena a lista dos sites a serem monitorados por mudanças e o resultado da aplicação da função hash no último acesso feito pelo agente de monitoramento;
- comunicação: este módulo implementa os mecanismos de comunicação com os demais agentes.

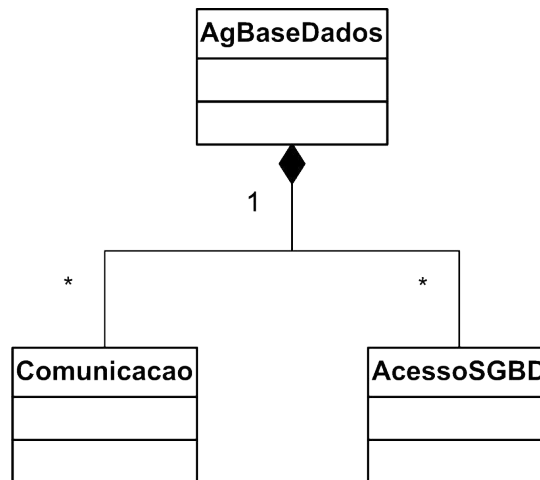


Figura 14 Proposta de estrutura para o agente de banco de dados

A estrutura proposta para o agente de base de dados pode ser vista na figura 14. Esta estrutura é bem simples, já que o papel do agente de base de dados é servir de interface entre os demais agentes e o sistema de banco de dados. Esta estrutura é composta do módulo de comunicação com os demais agentes e o módulo de acesso ao banco de dados, no qual são implementadas as técnicas específicas do sistema de banco de dados escolhido. Esta modularidade permite que se troque o sistema de banco de dados sem que sejam necessárias muitas mudanças no sistema de agentes como um todo, somente no módulo de acesso aos dados do agente de base de dados (ECKEL, 2002; MEYER, 2000).

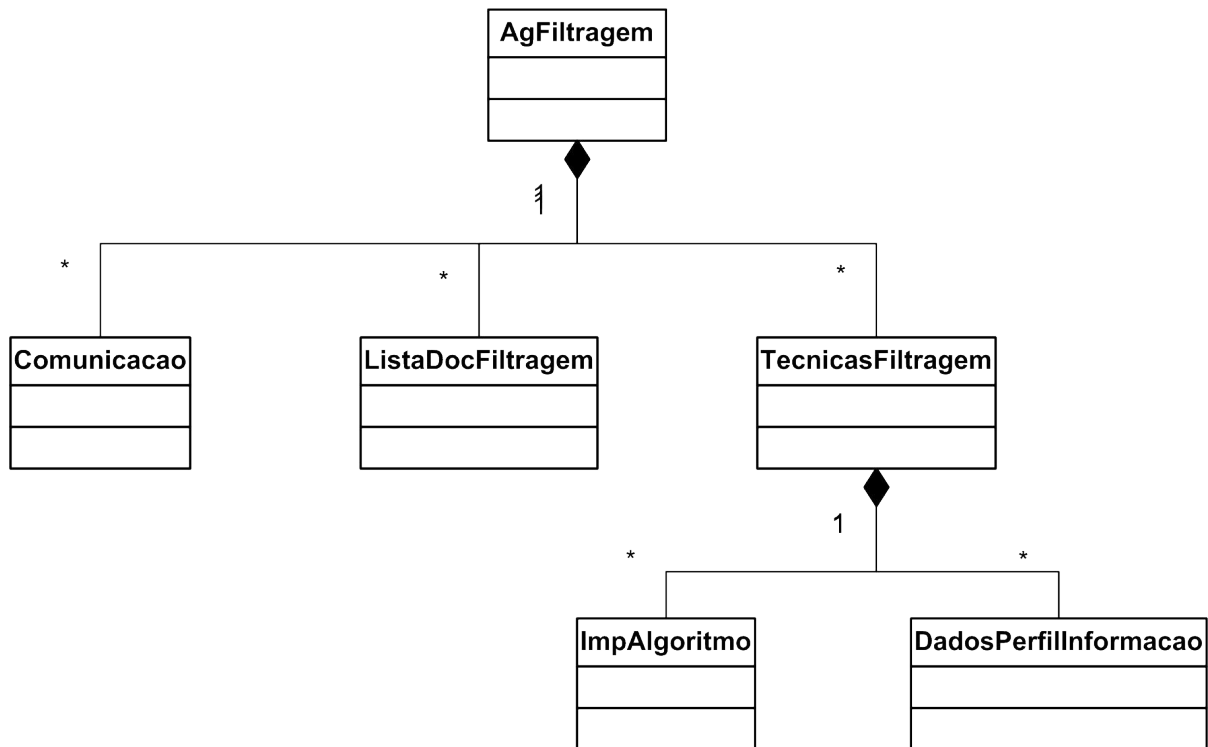


Figura 15 Proposta de estrutura para os agentes de filtragem

Os agentes de filtragem, que implementam as funcionalidades de análise de texto, possuem uma estrutura modular representada na figura 15. Os diversos tipos de agentes de filtragem compartilham uma mesma estrutura, com as diferenças se traduzindo em diferentes classes implementando um módulo, utilizando os princípios de orientação a objetos (ECKEL, 2002; MEYER, 2000). Os módulos que compõem os agentes de filtragem são:

- comunicação: responsável por implementar a comunicação com os demais agentes;
- lista de documentos a analisar: chamado ListaDocFiltragem na figura 15, este módulo armazena os documentos a serem processados pelo módulo de análise de texto;
- módulo de análise de textos: chamado TecnicasFiltragem na figura 15, este módulo, composto por dois outros módulos, implementa as técnicas de

inteligência artificial que analisam o texto dos documentos. É composto pelo módulo `ImpAlgoritmo`, que implementa o algoritmo de análise de texto específico do tipo de agente, e do módulo `DadosPerfilInformacao`, que armazena as informações necessárias para o funcionamento do algoritmo, entre elas palavras chave, parâmetros de configuração do algoritmo e informações usadas na execução do mesmo.

As técnicas de mineração de texto, discutidas na seção 3.7, no atual estágio de desenvolvimento da proposta, não foram implementadas ou especificadas. A seleção dos algoritmos adequados deve ser feita a medida que as várias técnicas forem sendo implementadas nos respectivos módulos.

Na próxima seção, serão apresentadas e discutidas as mensagens trocadas pelos agentes.

## ***4.5 Mensagens entre agentes***

Nesta seção serão apresentadas algumas das mensagens trocadas pelos agentes. Como a proposta para o modelo do sistema é muito grande e complexa, muitas das mensagens não foram especificadas, o foco sendo mantido nas informações trocadas pelos agentes que são essenciais para o entendimento de suas tarefas no modelo proposto. As mensagens estão representadas em diagramas UML, nos quais os agentes aparecem como objetos (instâncias) das classes de agentes propostas na seção anterior. As mensagens aparecem como linhas direcionais, indicando o agente que origina a mensagem e o conteúdo informacional principal

que a mensagem transporta.

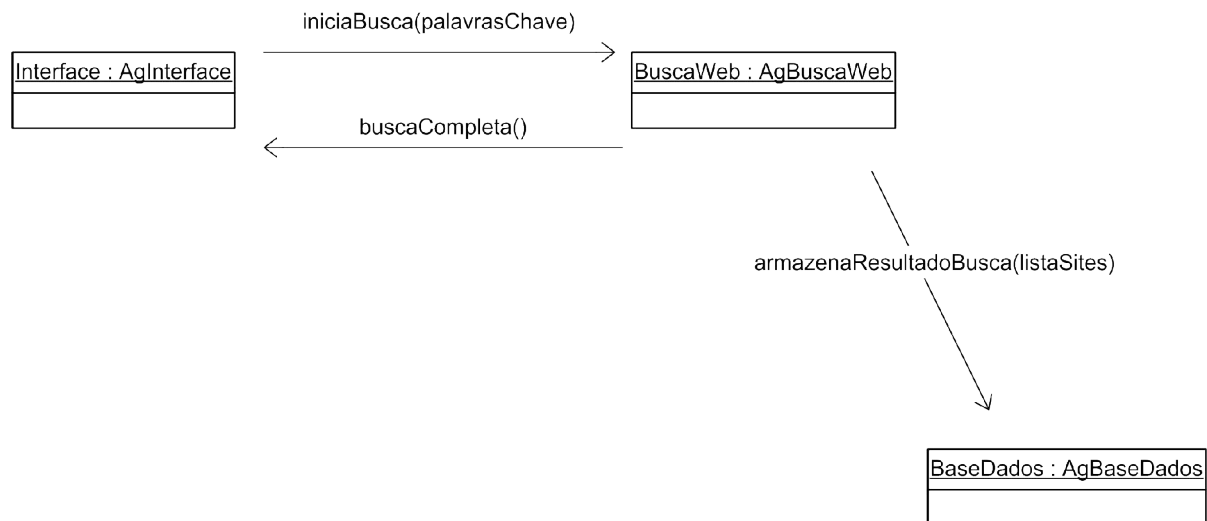


Figura 16 Algumas mensagens chave na atividade de busca na Web

A primeira figura, figura 16, representa as mensagens trocadas no caso de uma atividade de busca na Web. Os agentes envolvidos nesta atividade são o agente de interface, chamado Interface, o agente de busca na Web, chamado BuscaWeb, e o agente de banco de dados, chamado BaseDados. Quando, em resposta à requisição do usuário, o agente de interface ativa o processo de busca, este envia uma mensagem, chamada **iniciarBusca()**, ao agente de busca. Esta mensagem contém como informação chave um conjunto de palavras-chave a serem usadas na busca. Após o recebimento desta mensagem, o agente de busca irá realizar a busca pedida, e quando terminada, enviará uma mensagem ao agente de base de dados, chamada **armazenaResultadoBusca()**, com a lista de sites retornados como o principal conteúdo informacional desta mensagem. Após o envio desta mensagem, o agente de busca envia uma mensagem ao agente de interface, chamada **buscaCompleta()** na figura, para informá-lo de que a busca foi terminada.

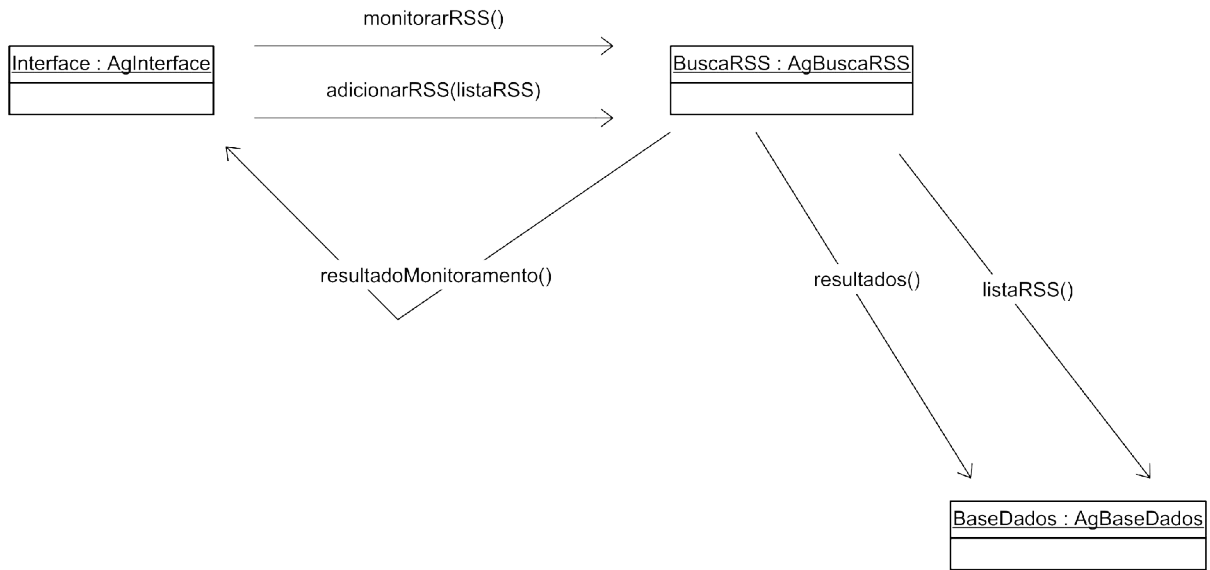


Figura 17 Algumas mensagens na atividade de busca em RSS

As mensagens trocadas quando da busca em um site que disponibilize RSS são mostradas na figura 17. Os agentes que participam desta busca são o agente de interface (Interface), o agente de banco de dados (BaseDados) e o agente de busca (BuscaRSS). A busca em um site que disponibilize RSS envolve o envio de uma mensagem pelo agente de interface, chamada `monitorarRSS()`, para iniciar o processo de monitoramento, ou a mensagem `adicionarRSS()`, para adicionar um site RSS aos já monitorados pelo agente. Estas mensagens têm como informação central uma lista de sites que disponibilizem RSS. A intervalos periódicos, o agente de busca envia para o agente de banco de dados mensagem contendo a lista de sites a serem monitorados para armazenamento, chamada `listaRSS()`, e mensagem contendo o resultado das buscas realizadas, chamada `resultados()`. O agente de busca também envia mensagens periódicas ao agente de interface para informar do progresso da atividade de busca, chamadas `resultadoMonitoramento()`.

A tarefa de monitorar sites cadastrados está representada na figura 18. Ela envolve o agente de interface (Interface), o agente de monitoramento (Monitor) e o agente de banco de dados (BaseDados). O agente de interface envia mensagens para adicionar ou remover sites da lista de sites a serem monitorados, estas mensagens são, respectivamente, `adicionarSite()` e

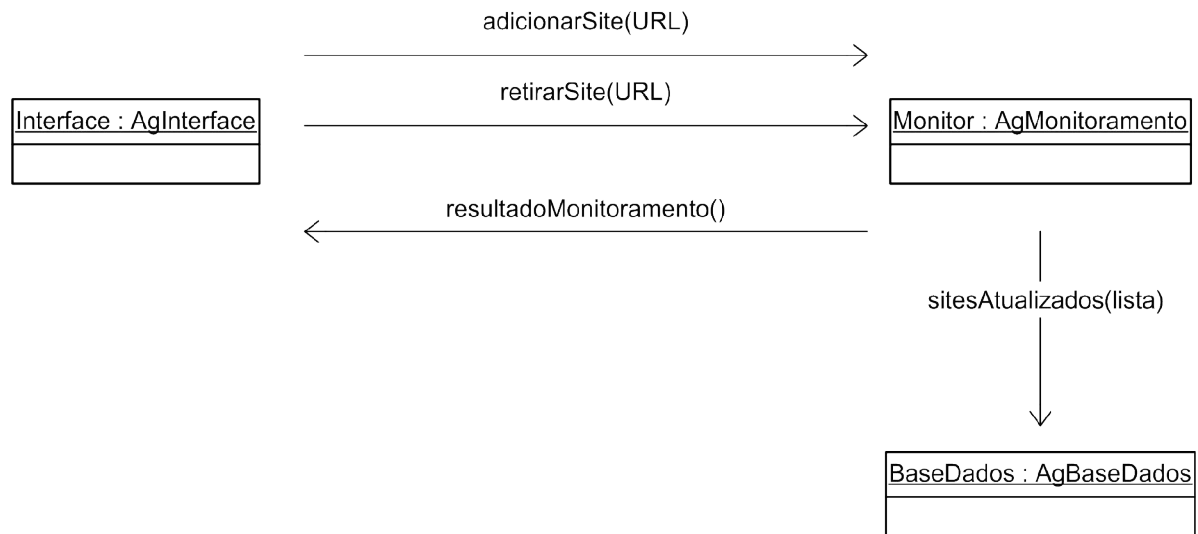


Figura 18 Algumas mensagens na atividade de monitoramento

`retirarSite()`. O principal conteúdo destas mensagens é o endereço, ou URL, do site a ser monitorado. Quando o agente de monitoramento termina uma atividade, envia ao agente de banco de dados uma mensagem, chamada na figura de `sitesAtualizados()`, que contém a lista dos sites que foram atualizados desde a última visita do agente. Também envia uma mensagem ao agente de interface, chamada `resultadoMonitoramento()`, que informa ao agente de interface que a atividade foi completada.

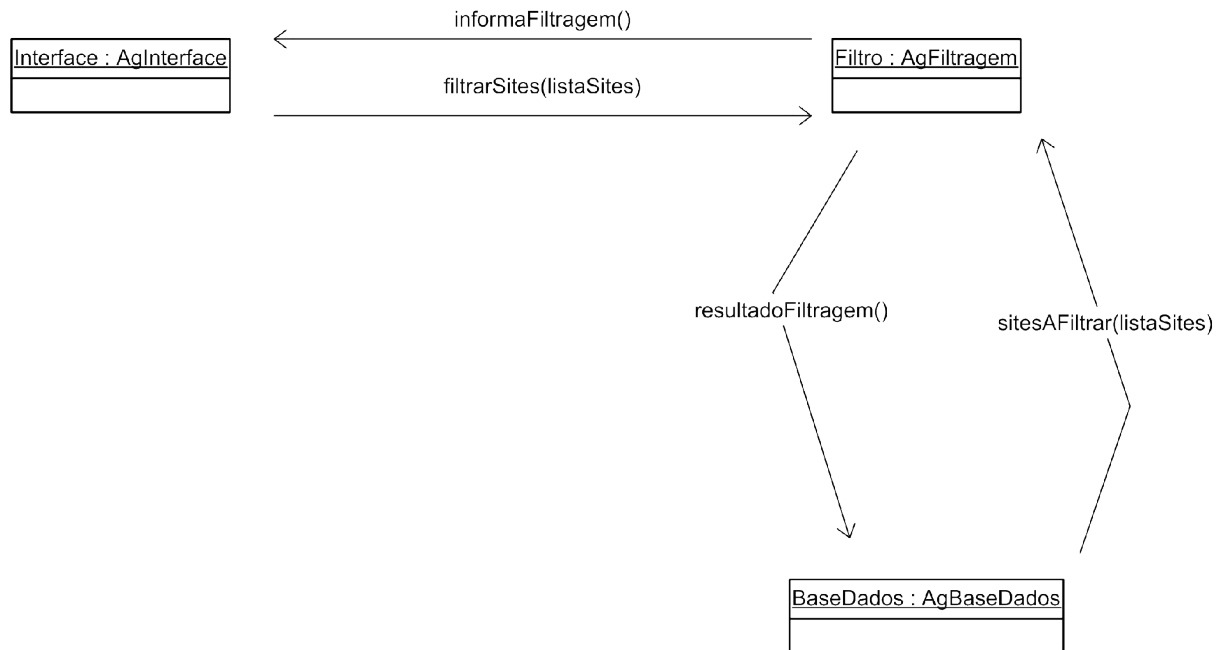


Figura 19 Algumas mensagens na atividade de filtragem

A tarefa de filtragem, representada na figura 19, envolve os agentes de interface (Interface), de filtragem (Filtro) e de banco de dados (BaseDados). O agente de interface envia uma mensagem para o agente de filtragem, chamada `filtrarSites()`, contendo uma lista dos sites a serem filtrados. O agente de filtragem prossegue então para realizar a filtragem dos sites utilizando a técnica de análise que implementa, buscando o conteúdo dos mesmos do banco de dados através da mensagem chamada `sitesAFiltrar()`, na figura. Após a atividade completa, o agente de filtragem envia uma mensagem ao agente de banco de dados contendo os resultados para serem armazenados, chamada `resultadoFiltragem()`, e envia uma mensagem ao agente de interface para informar que a atividade foi cumprida, chamada `informaFiltragem()`.



## 5 Conclusão

O presente trabalho tratou da proposta de modelo de uma arquitetura de agentes inteligentes para o monitoramento de fontes de informação na internet, para o uso como ferramenta de apoio ao processo de Inteligência Competitiva.

Partindo de uma breve análise do que é necessário para a competitividade, seguindo para um estudo sobre a Inteligência Competitiva, o que ela é e sua importância estratégica para as organizações, estabeleceu-se a importância da Internet, em especial a World Wide Web, como fonte de informações de baixo custo.

Estabeleceu-se então que a Web é um território amplo e não estruturado, que requer muito tempo para um usuário simplesmente monitorar sites conhecidos em busca de atualizações, e ainda mais para a busca de novos sites que contenham informações relevantes. A partir desta constatação, identificou-se a necessidade de uma ferramenta que automatizasse as tarefas mais repetitivas da etapa de coleta no processo de Inteligência Competitiva.

Foi realizada um estudo da teoria de agentes inteligentes, suas características, capacidades e arquiteturas. Este estudo serviu como base para a definição dos componentes da ferramenta, assim como da separação das diversas atividades envolvidas na coleta de informações entre os componentes.

Tomou-se como base o modelo proposto por Silva (2000) para a proposta feita neste trabalho. Esta proposta, descrita na página 70, consiste da descrição dos agentes que compõem o modelo proposto, seguida da interação do modelo proposto com o usuário. Uma descrição mais detalhada da proposta para cada tipo de agente então se segue, com suas funções e capacidades. Sua estrutura então é abordada, seguindo-se pela descrição das mensagens principais trocadas entre os agentes.

A contribuição do presente trabalho ao estudo da Inteligência Competitiva é aprofundar a discussão sobre o papel da Internet no processo, e contribuir para a criação de uma ferramenta que permita que organizações executem o processo de Inteligência Competitiva de modo efetivo sem uma necessidade muito elevada de recursos financeiros. A utilidade da Internet no processo de Inteligência Competitiva ainda não foi totalmente realizada devido à dificuldade de encontrar informações importantes em um universo tão vasto e caótico, e pretende-se contribuir para o desenvolvimento de procedimentos, metodologias e técnicas que aumentem o desempenho da Internet neste processo.

A descrição dos diversos agentes e suas habilidades dá margem para outras pesquisas na área, desenvolvendo ferramentas para tratar texto não estruturado de maneiras a gerar resultados com real valor para os profissionais da área de Inteligência Competitiva.

## **5.1 Limitações**

Foi realizado um estudo teórico resultando na proposta de um modelo de uma arquitetura de um sistema multiagentes. O foco deste estudo foi na interação entre os vários

agentes necessários para o cumprimento de suas tarefas, especificamente nos dados necessários para cada agente, e que resultado seria obtido de sua operação. Também foi feita uma descrição das diversas tecnologias de análise de textos não estruturados a serem empregadas no modelo proposto. Pelo fato de o estudo ser uma descrição teórica e abstrata, não chegou-se a uma especificação detalhada dos agentes e seus componentes, a ser usada para uma eventual implementação do modelo, focalizando-se os esforços em uma proposta de funcionalidade e de soluções que atendam às necessidades do subprocesso de coleta de informações do processo de inteligência competitiva.

## ***5.2 Sugestões para estudos futuros***

A partir da proposta de modelo da arquitetura, pode-se dar continuidade ao trabalho através da modelagem detalhada dos diversos agentes e do sistema com um todo. Como propôs-se um modelo inerentemente modular, a tarefa de modelagem e implementação dos agentes pode ser realizada em diferentes momentos, especialmente os diversos agentes de filtragem.

Como a proposta prevê a existência de vários agentes trabalhando com algoritmos distintos em uma mesma tarefa, um sistema resultante deste modelo serviria não somente para a aplicação no processo de Inteligência Competitiva, mas também como um banco de testes de diferentes algoritmos de mineração de textos a serem experimentados em situação real.

Outra tarefa que é necessária para a implementação do modelo é a criação e treinamento de um sistema de análise de linguagem natural em língua portuguesa, que possa

ser usado em texto não estruturado. Esta tarefa em si representa um esforço muito elevado, dada a complexidade da língua e a ambigüidade inerente a documentos em um domínio aberto.

## Referências

ABOUT THE WORLD WIDE WEB. Disponível em <http://www.w3.org/WWW/>. Acesso em maio 2005

ABREU, Aline F.; ABREU, Pedro F.. **Apostila de GTI**. Núcleo de Estudos em Inovação, Gestão e Tecnologia de Informação. Florianópolis, 2003.

AH-HWEE Tan. **Text Mining**: The state of the art and the challenges. In Proceedings, PAKDD'99 workshop on Knowledge Discovery from Advanced Databases, Beijing, April 1999, p65-70.

BERNHARDT, Douglas. **Competitive Intelligence**; How to acquire and use corporate intelligence and counter-intelligence. New York: Prentice Hall, Financial Times, 2003

CAIRE, G. **Jade Tutorial**, 2002, Disponível em <http://jade.tilab.com/>

CARDOSO, Olinda N. P.. **Recuperação de Informação**. In: SEMANA DE CIÊNCIA DA COMPUTAÇÃO DA UFLA, 2000, Lavras. Infocomp. 2000.

CARDOSO, Walter Felix Junior. **A Inteligência Competitiva Aplicada nas Organizações do Conhecimento como Modelo de Inteligência Empresarial Estratégica para Implementação e Gestão de Novos Negócios**. 2003. Tese - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis

CARLEY, Kathleen M.; GASSER, Les. **Computational Organization Theory**. In: Multiagent Systems, a Modern Approach to Distributed Artificial Intelligence. Cambridge, 1999

CENDÓN, B. V. **Ferramentas de busca na Web**. Ci. Inf., Brasília, v. 30, n. 1, p. 39-49, jan./abr. 2001

CHAKRABARTI, S., **Mining the Web**: discovering knowledge from hypertext data. Morgan Kaufmann Publishers, London, 2003.

CORMEN, T. H. et al. **Introduction to Algorithms**. 2. ed. Cambridge: The MIT Press, 2001

DECKER, Keith S., SYCARA, Katia. **Intelligent Adaptive Information Agents**, Journal of Intelligent Information Systems, Kluwer Academic Publishers, Boston. 1997

DETERS, J. I.; ADAIME, S. F. **Um estudo comparativo dos sistemas de busca na web**. Anais do V Encontro de Estudantes de Informática do Tocantins. Palmas, TO. outubro, 2003. pp. 189-200

DURFEE, Edmund H.. **Distributed Problem Solving and Planning**. In: Multiagent Systems, a Modern Approach to Distributed Artificial Intelligence. Cambridge: MIT Press, 1999

EBECKEN, Nelson F. F.; LOPES, Maria Celia S.; COSTA, Myrian C. A.. **Mineração de Textos**. In: Sistemas Inteligentes Fundamentos e Aplicações. Barueri, SP:Manole, 2003

ECKEL, Bruce. **Thinking in Java**. 3. ed. MindView, Inc. 2002. Disponível em: <http://mindview.net/Books>

FRANKLIN, S.; GRAESSER, A.. **Is it an Agent, or just a Program?:A Taxonomy for Autonomous Agents**. Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, 1996

GARCIA, Ana Cristina B.; SICHMAN, Jaime S. **Agentes e Sistemas Multiagentes**. In: Sistemas Inteligentes Fundamentos e Aplicações. Barueri, SP:Manole, 2003

GOOGLE WEB APIs., 2005. Disponível em: <http://www.google.com/apis/index.html>. Acesso em: Maio, 2005

Haykin, Simon. **Neural Networks**, A Comprehensive Foundation. IEEE Press, 1994

HIDDEN Markov Model., 2005. Disponível em : [http://en.wikipedia.org/wiki/Hidden\\_Markov\\_Model](http://en.wikipedia.org/wiki/Hidden_Markov_Model). Acesso em: Maio, 2005

HUHNS, Michael N.; STEPHENS, Larry M.. **Multiagent Systems and Societies of Agents**. In: Multiagent Systems A Modern Approach to Distributed Modern Approach to Artificial Intelligence. Cambridge MIT Press, 1999

JADE. <http://jade.tilab.com/>. Acessado em 2005

JUNGES, Ivone. **Metodologia para a Identificação de Cenários Tecnológicos de Pequenas e Médias Empresas que Atuam em Redes Interempresariais do Tipo Topdown**. 2004. Tese - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis

KOBAYASHI, Mei; AONO, Masaki. **Vector Space Models for Search and Cluster Mining**. In: Survey of Text Mining, Clustering, Classification and Retrieval. Springer, 2004

KRÖSE, Ben; VAN DER SMAGT, Patrick. **An Introduction to Neural Networks**. University of Amsterdam, 1996

LAFORE, R. **Data Structures and Algorithms in Java**. Waite Group Press 1998

LIEBERMAN, H.; MAULSBY, D. **Instructible agents**: Software that just keeps getting better, IBM SYSTEMS JOURNAL, 1996

LIEBERMAN, Henry. **Art Imitates Life**: Programming by Example as an Imitation Game. ,, 2002

LIEBERMAN, Henry. **Programming by Example Home Page**. 2005. Disponível em: <http://web.media.mit.edu/~lieber/PBE/>. Acesso em:

LIU, Jiming. **Autonomous Agents and Multi-Agent Systems**. London: World Scientific, 2001

LUSTOSA, Volney G.. **O Estado da Arte em Inteligência Artificial**. 2004 Dissertação - Bacharelado de Sistemas de Informação, Universidade Católica de Brasília,

MARCO, Sueli A.. **Inteligência Competitiva**: Definições e Contextualização. Revista Transinformação, Campinas, 1999

MARTINS, Jefferson G.. **Uma arquitetura baseada em agentes inteligentes para ambientes computacionais voltados à educação a distância**. 2002 Dissertação - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis

Meyer, B. **Object-Oriented Software Construction** Prentice. Hall PTR, 2000

OLIVEIRA, Álvaro B.. **Um modelo baseado em agentes para o processo de coleta de dados via WEB**. 2000. Tese - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis

RAYNOR, William J. Jr.. **The International Dictionary of Artificial Intelligence**. Chicago: Glenlake Publishing Company, Ltd., 1999

RSS Tutorial., 2005. Disponível em: <http://www.w3schools.com/rss/default.asp>. Acesso em: Maio, 2005

RUSSEL, Peter; NORVIG, Stuart. **Artificial Intelligence A Modern Approach**. Englewood Cliffs: Prentice Hall, 1995

SILVA, E. L.; MENEZES, E. M. **Metodologia da pesquisa e elaboração de dissertação**. – 4. ed. – Florianópolis: UFSC, 2005.

SILVA, Helena P.. **Inteligência Competitiva na Internet**; Proposta de um Processo. 2000. Tese - Programa de Pós Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis

FININ, Tim, et al. **Specification of the KQML Agent-Communication Language**. 1993. Disponível em: <http://www.cs.umbc.edu/kqml/kqmlspec/spec.html>. Acesso em: Março, 2005

SPERB, Rafael M.. **Agentes inteligentes difusos**: uma ferramenta híbrida para exploração de processos espaciais em zonas costeiras. 2002. Tese - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis

THIRY, M.. **Uma arquitetura baseada em agentes para suporte ao ensino à distância**. 1999. Tese - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis

TYSON, Kirk W. M.. **The Complete Guide to Competitive Intelligence**. Lisle Chicago: Kirk Tyson International, 1998



WATSON, Mark. **Practical Artificial Intelligence Programming in Java**.  
www.markwatson.com, 2004

WHAT Is CI?. 2004. Disponível em:<http://www.scip.org/ci/>. Acesso em:Fevereiro, 2005

WOOLDRIDGE, Michael; JENNINGS, Nicholas R.. **Intelligent Agents**: Theory and Practice. 1994

WOOLDRIDGE, Michael. **Intelligent Agents**. In: Multiagent Systems A Modern Approach to Distributed Artificial Intelligence. Cambridge:The MIT Press, 1999

YAHOO! Search Developer Network.., 2005. Disponível em:<http://developer.yahoo.net/>. Acesso em: Maio, 2005

YANG, Yiming. **An Evaluation of Statistical Approaches to Text Categorization**. Information Retrieval, 1999